How can I ...

Connect PlantStruxure to FOUNDATION Fieldbus?

System Technical Note

Optimized functional unit

Design

Your architecture

Schneider Electric

Important Information

People responsible for the application, implementation and use of this document must make sure that all necessary design considerations have been taken into account and that all laws, safety and performance requirements, regulations, codes, and applicable standards have been obeyed to their full extent.

Schneider Electric provides the resources specified in this document. These resources can be used to minimize engineering efforts, but the use, integration, configuration, and validation of the system is the user's sole responsibility. Said user must ensure the safety of the system as a whole, including the resources provided by Schneider Electric through procedures that the user deems appropriate.

Notice

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the safety requirements for the equipment used in their systems, or compliance with both national or international safety laws and regulations.

Readers are considered to already know how to use the products described in this document.

This document does not replace any specific product documentation.

The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER
DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury. **Failure to follow these instructions will result in death or serious injury.**

⚠ WARNING
WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.
Failure to follow these instructions can cause death, serious injury or equipment damage.

⚠ CAUTION
CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.
Failure to follow these instructions can result in injury or equipment damage.

NOTICE
NOTICE is used to address practices not related to physical injury.
Failure to follow these instructions can result in equipment damage.

Note: Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction, operation and installation of electrical equipment, and has received safety training to recognize and avoid the hazards involved.

Before You Begin

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions and government regulations etc. In some applications more than one processor may be required when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation and maintenance of the solution. Therefore only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to

the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual also provides much useful information.

Ensure that appropriate safeties and mechanical/electrical interlocks protection have been installed and are operational before placing the equipment into service. All mechanical/electrical interlocks and safeties protection must be coordinated with the related automation equipment and software programming.

Note: Coordination of safeties and mechanical/electrical interlocks protection is outside the scope of this document.

START UP AND TEST

Following installation but before using electrical control and automation equipment for regular operation, the system should be given a start up test by qualified personnel to verify the correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ WARNING
EQUIPMENT OPERATION HAZARD • Follow all start up tests as recommended in the equipment documentation. • Store all equipment documentation for future reference. • Software testing must be done in both simulated and real environments. **Failure to follow these instructions can cause death, serious injury or equipment damage.**

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the USA, for example). If high-potential voltage testing is necessary, follow recommendations in the equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment
- Close the equipment enclosure door
- Remove ground from incoming power lines
- Perform all start-up tests recommended by the manufacturer

OPERATION AND ADJUSTMENTS

The following precautions are from NEMA Standards Publication ICS 7.1-1995 (English version prevails):

Regardless of the care exercised in the design and manufacture of equipment or in the selection and rating of components; there are hazards that can be encountered if such equipment is improperly operated.

It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.

Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

⚠ WARNING

UNEXPECTED EQUIPMENT OPERATION

- Only use software tools approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can cause death, serious injury or equipment damage.

INTENTION

This document is intended to provide a quick introduction to the described system. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers information additional to the product documentation on installation, configuration and implementing the system.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider Electric and third-party components may be integrated to fulfill an industrial application.

A detailed functional description or the specifications for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The architecture described in this document has been fully tested in our laboratories using all the specific references you will find in the component list near the end of this document. Of course, your specific application requirements may be different and will require additional and/or different components. In this case, you will have to adapt the information provided in this document to your particular needs. To do so, you will need to consult the specific product documentation of the components that you are substituting in this architecture. Pay particular attention in conforming to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

It should be noted that there are some major components in the architecture described in this document that cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified herein. You must be aware of the consequences of component substitution in the architecture described in this document as substitutions may impair the compatibility and interoperability of software and hardware.

⚠ CAUTION
EQUIPMENT INCOMPATIBILITY OR INOPERABLE EQUIPMENT Read and thoroughly understand all hardware and software documentation before attempting any component substitutions. **Failure to follow these instructions can result in injury or equipment damage.**

This document is intended to describe how to connect PlantStruxure to FOUNDATION Fieldbus.

⚠ DANGER

HAZARD OF ELECTRIC SHOCK, BURN OR EXPLOSION

- Only qualified personnel familiar with low and medium voltage equipment are to perform work described in this set of instructions. Workers must understand the hazards involved in working with or near low and medium voltage circuits.
- Perform such work only after reading and understanding all of the instructions contained in this bulletin.
- Turn off all power before working on or inside equipment.
- Use a properly rated voltage sensing device to confirm that the power is off.
- Before performing visual inspections, tests, or maintenance on the equipment, disconnect all sources of electric power. Assume that all circuits are live until they have been completely de-energized, tested, grounded, and tagged. Pay particular attention to the design of the power system. Consider all sources of power, including the possibility of back feeding.
- Handle this equipment carefully and install, operate, and maintain it correctly in order for it to function properly. Neglecting fundamental installation and maintenance requirements may lead to personal injury, as well as damage to electrical equipment or other property.
- Beware of potential hazards, wear personal protective equipment and take adequate safety precautions.
- Do not make any modifications to the equipment or operate the system with the interlocks removed. Contact your local field sales representative for additional instruction if the equipment does not function as described in this manual.
- Carefully inspect your work area and remove any tools and objects left inside the equipment.
- Replace all devices, doors and covers before turning on power to this equipment.
- All instructions in this manual are written with the assumption that the customer has taken these measures before performing maintenance or testing.

Failure to follow these instructions will result in death or serious injury.

The STN Collection

The implementation of an automation project includes five main phases: Selection, Design, Configuration, Implementation and Operation. To help you develop a project based on these phases, Schneider Electric has created the Tested, Validated, Documented Architecture and System Technical Note.

A Tested, Validated, Documented Architecture (TVDA) provides technical guidelines and recommendations for implementing technologies to address your needs and requirements, This guide covers the entire scope of the project life cycle, from the Selection to the Operation phase, providing design methodologies and source code examples for all system components.

A System Technical Note (STN) provides a more theoretical approach by focusing on a particular system technology. These notes describe complete solution offers for a system, and therefore support you in the Selection phase of a project. The TVDAs and STNs are related and complementary. In short, you will find technology fundamentals in an STN and their corresponding applications in one or several TVDAs.

Development Environment

Each TVDA or STN has been developed in one of our solution platform labs using a typical PlantStruxure architecture.

PlantStruxure, the process automation system from Schneider Electric, is a collaborative architecture that allows industrial and infrastructure companies to meet their automation needs while at the same time addressing their growing energy efficiency requirements. In a single environment, measured energy and process data can be analyzed to yield a holistically optimized plant.

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this document is to provide information to build basic knowledge about FOUNDATION Fieldbus and how to connect it to a PlantStruxure architecture.

This document provides a simple example of a H1 FOUNDATION Fieldbus segment and how the instruments connected to the FOUNDATION Fieldbus communicate with a Schneider Electric Quantum PAC. It also explains how to use Unity Pro as an asset management tool for the instruments. Finally, a SCADA project is built to complete the architecture.

The FOUNDATION Fieldbus instrumentation is integrated using a gateway designed by SOFTING, which is member of a Schneider Electric's Collaborative Automation Partner Program (CAPP). CAPP is a formal community of business partners through which Schneider Electric expands its capabilities. This document also includes instrumentation from our partner KROHNE.

Although FOUNDATION Fieldbus H1 segments are suitable for hazardous environment installations, this document is not intended to describe and provide recommendations about the installation in hazardous areas. Therefore, the project described in this document does not take in account the constraints and regulations related to hazardous environments.

1.2. Prerequisites

For better understanding of this document, we recommend knowledge of the following software:

- Unity Pro
- OFS
- Vijeo Citect

1.3. Glossary

A glossary is available in the appendix chapter of this document. Please refer to it whenever necessary.

2. FOUNDATION Fieldbus overview

2.1. Introduction

FOUNDATION Fieldbus is a bus communication system designed for the following:

- Process control
- Field instrumentation monitoring

The following process control examples are continuous processes:

- Pipeline flow control
- Tank level control
- Temperature control

These processes are commonly found in oil and gas industries or petrochemical plants, among others.

2.1.1. H1 and HSE

The FOUNDATION Fieldbus standard defines two communication protocols that use different physical media and communication speeds:

- **H1** mainly connects the field devices, providing communication and power to the devices (such as sensors, actuators and controllers) using the Manchester-encoded Bus Powered (MBP) technology over a twister-pair wiring with a data flow of at 31.25 kbps.
- **High Speed Ethernet (HSE)** uses a 10/100 Mbps Ethernet as a physical layer, providing a high-speed backbone for the network. This is commonly used to provide interoperability between supervision systems and the instruments connected to a H1 segment. Specific devices are designed for HSE and can be connected to the HSE directly.

H1 and HSE were specifically designed as complementary networks. H1 is optimized for traditional process control applications. On the other hand, HSE, which employs commercial off-the-shelf Ethernet equipment (such as switches, routers and firewalls), is designed for high-performance control applications and plant information integration.

The physical link between the H1 segments and HSE networks is created using special communication linking devices. These devices will be discussed later in this document.

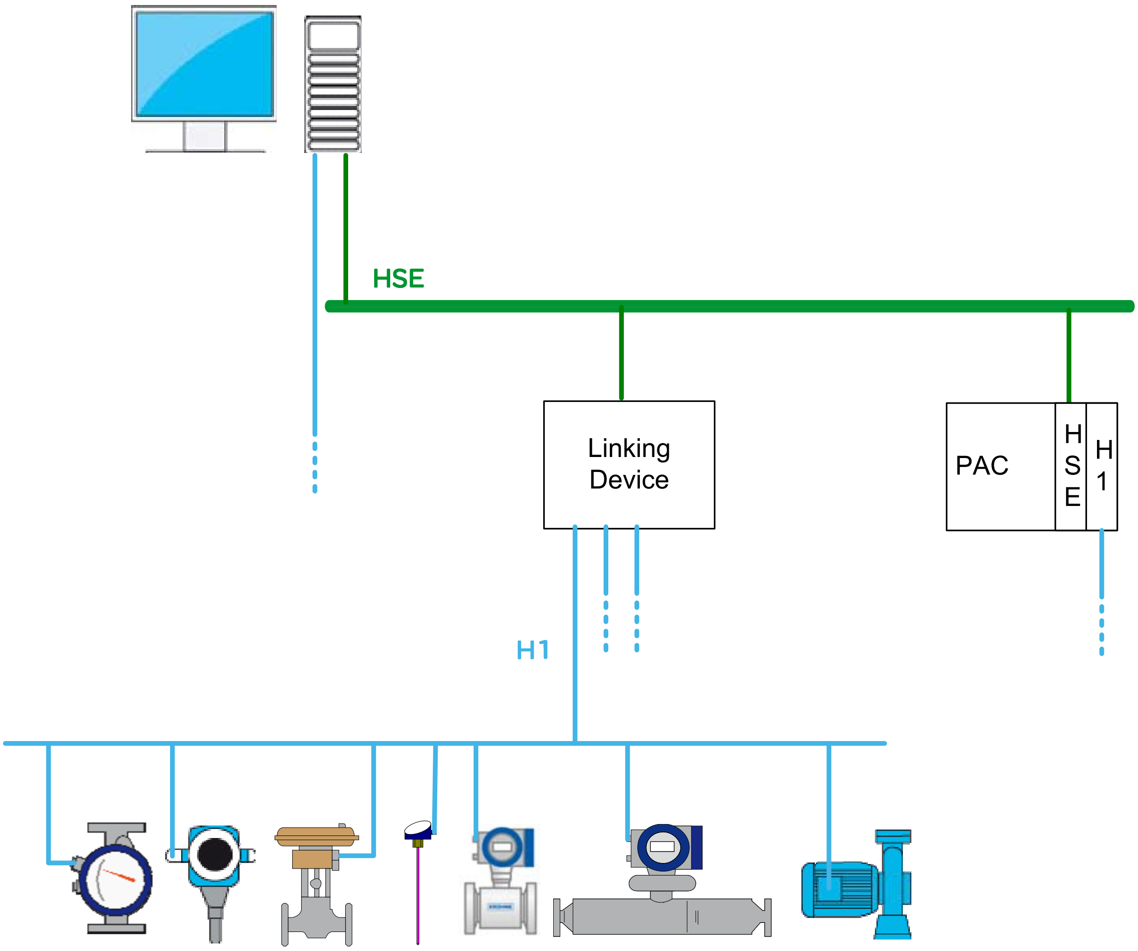


Figure 1: FOUNDATION Fieldbus-based control system

2.1.2. H1 and hazardous areas

The H1 bus can be designed as intrinsically safe (IS) to suit applications in hazardous areas. To fulfill the hazardous areas requirements, proper barriers and terminators should be installed between non-hazardous and hazardous areas. Furthermore, all the hardware (including the instrumentation) must be certified and labeled according to the targeted hazardous area. Depending on the hazardous area, the capacity of the electrical lines in the bus is limited. Accordingly, the bus distances and the number of devices that can be connected to one segment are possibly reduced.

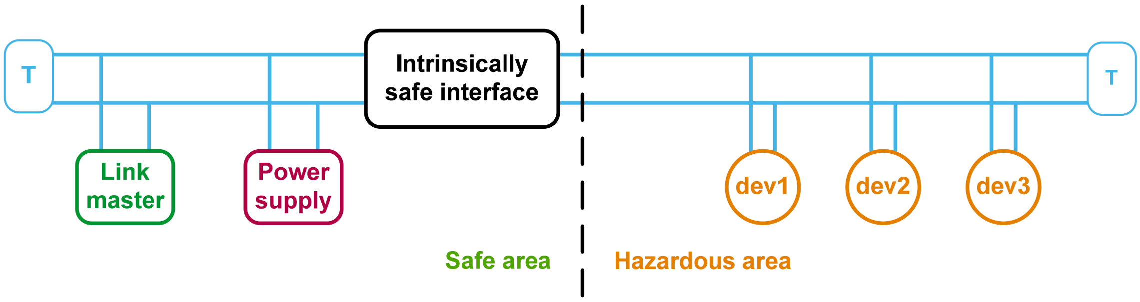


Figure 2: FOUNDATION Fieldbus in hazardous areas

Since the FOUNDATION fieldbus specification is not based on the FISCO model, the plant operator must ensure that intrinsic safety requirements are met when planning and installing the communications network. In any case, equipment should only be installed, operated, serviced and maintained by qualified personnel.

⚠ DANGER

HAZARD OF ELECTRIC SHOCK, BURN OR EXPLOSION

This guide is not intended to describe rules for the development of an automation system in an explosive environment.

If you design an automation system in an explosive environment, you must follow the standards and rules which are relevant to the countries or regions where the system is designed and installed.

Failure to follow these instructions will result in death or serious injury.

2.1.3. H1 notions

This subsection presents the concepts of the H1 architectures.

Segment

A FOUNDATION Fieldbus network is made up of devices connected by a serial bus. This serial bus is called a link or more commonly known as a segment. A fieldbus network consists of one or more segments and each segment is configured with a unique segment identifier.

Devices

Devices are identified on the fieldbus network by a tag (character string), a node ID (address on the H1 bus) and the device ID (serial number of the device). The tag and node ID can be customized by the user while the device ID is a manufacturer parameter that can not be modified. Three types of devices exist:

- Link master: A link master device is capable of controlling the communications traffic on a segment by scheduling the communication on the network. Every Fieldbus network needs at least one device with link master capabilities.
- Basic device: A basic device can not control the traffic on a segment. Typically, this kind of device is an instrument.
- H1 bridge: A bridge connects segments together to allow data transfer between different segments.

Blocks

The blocks can be thought of as the modeling of the functionalities and device data by the FOUNDATION Fieldbus standard. This applies to all the functions that can be present in a control process.

Three types of block are defined by the FOUNDATION Fieldbus standard:

- Resource Block
- Transducer Block
- Function Block

This topic will be covered in more depth later in this document.

Linkage

The function blocks configured to control a process are linked as a logical connection by configuration objects inside the devices. It is therefore possible to send data from one block to another. A linkage is different from a link, in that a link is a physical wire pair that connects

devices on a fieldbus network, while a linkage is a logical connection that connects two function blocks.

Figure 3: Function blocks linked together

Loops

A loop is a group of function blocks connected by linkages that are executed in a configured rate – for example, a period of one second.

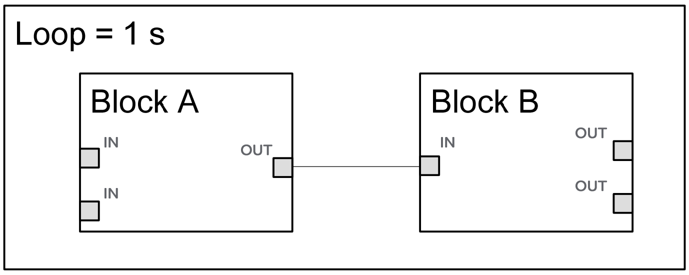


Figure 4: Function blocks application with one loop

It is possible to define several loops running at different rates for each segment, as illustrated in the following picture:

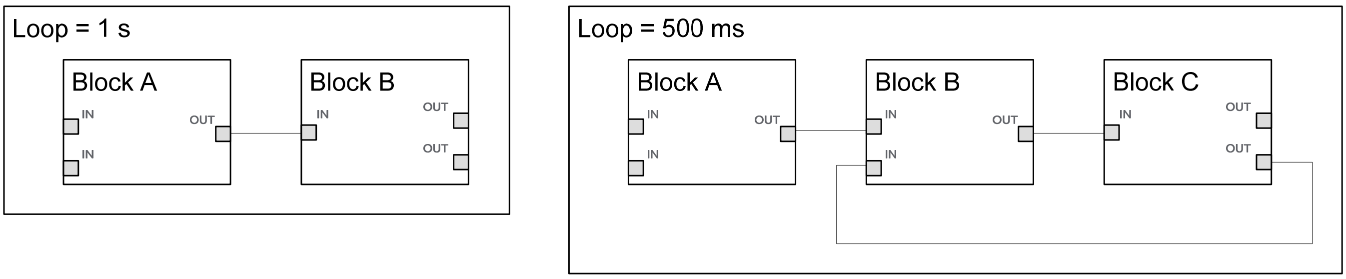


Figure 5: Function blocks application with two loops

2.1.4. HSE notions

This subsection presents the concepts of the HSE architectures.

HSE device

A HSE device is any FOUNDATION Fieldbus device connected directly to the HSE network.

HSE field device

A HSE field device is any device that contains at least one function block application process.

Linking device

A linking device is a HSE device that allows the connection of H1 segments to HSE networks. This then provides interoperability between the different physical networks, including the communication between two H1 segments.

I/O gateway device

An I/O gateway device is any HSE device which provides HSE access to non-FOUNDATION Fieldbus devices via function blocks.

2.1.5. Layers

The FOUNDATION fieldbus communication model is based on the ISO/OSI reference model. Layers three to six are not used, as you can see on the following figure:

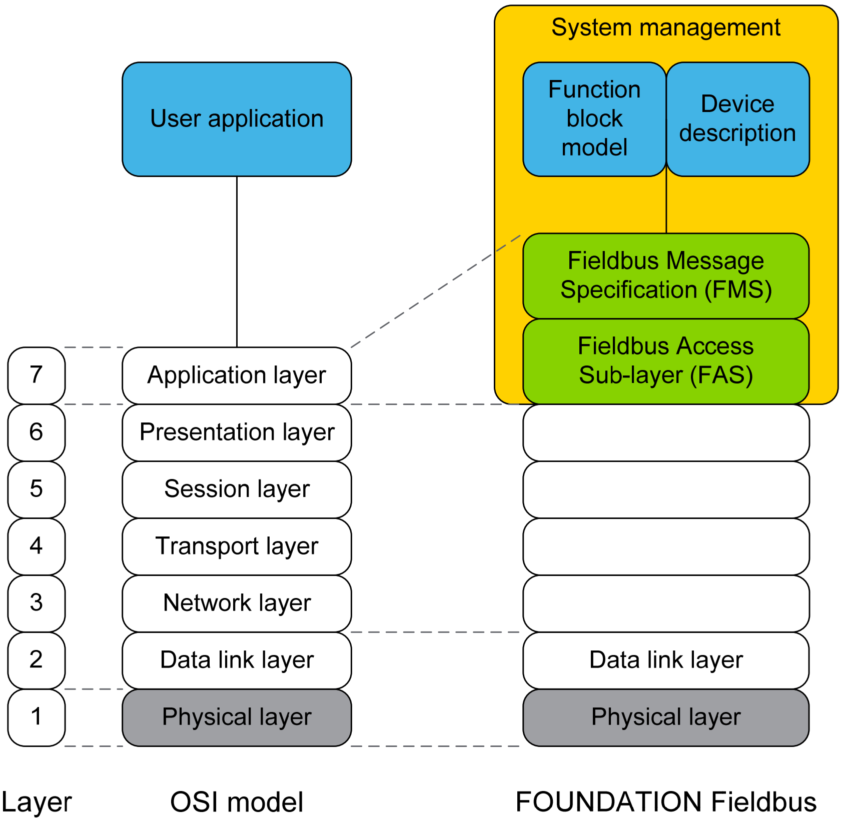


Figure 6: FOUNDATION Fieldbus layers description

The user application is made up of function blocks and the device description, and is directly based on the communication stack. Depending on which blocks are implemented in a device, users can access a variety of services. The services provided by the Fieldbus Access Sub-layer and Fieldbus Message Specification are transparent for the user.

2.2. Fieldbus components

The goal of this section is to provide an overview of the components of a H1 FOUNDATION Fieldbus architecture as described in the standard. The HSE network components are not discussed in this chapter because they use common network devices.

2.2.1. Power supply

The H1 segments use the MBP technology. In addition, a DC bus voltage (nominally 24 V but can be acceptable from 9 to 32 VDC) should be provided to each segment by a specific power supply. Furthermore, this power supply must be compliant with the isolation and power conditioning requirements in accordance with the FOUNDATION Fieldbus specification FF-831. As a result, all the DC power supplies that meet the minimum specifications have a FOUNDATION Fieldbus checkmark.

Figure 7: FOUNDATION Fieldbus registration checkmark logo.

⚠ DANGER
HAZARD OF ELECTRIC SHOCK, BURN OR EXPLOSION FOUNDATION Fieldbus (network and devices) must be powered by a FOUNDATION Fieldbus certified power supply. **Failure to follow these instructions will result in death or serious injury.**

When a FIELDBUS device is not transmitting, the device leaves the voltage waveform as a constant voltage (1). When the device transmits information, it modifies the signal to add a square wave signal to the 24 VDC (2), as described below:

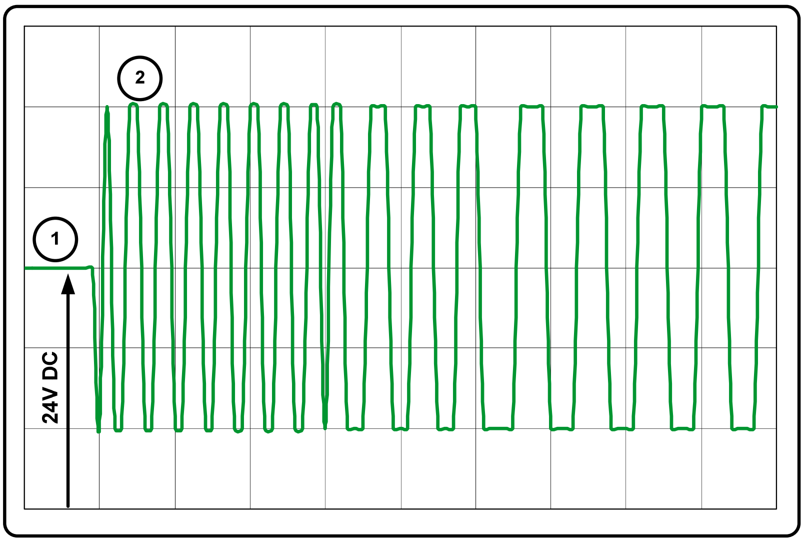


Figure 8: FOUNDATION Fieldbus electrical signal

Three types of power supplies are referenced in the FOUNDATION Fieldbus physical layer profile specification:

- TYPE 131: Non-IS power supply intended for feeding an IS barrier
- TYPE 132: Non-IS power supply and **not** intended for feeding an IS barrier
- TYPE 133: IS power supply, compliant with specific IS parameters

If they fulfill the physical layer standard requirements, power supplies can also provide redundancy.

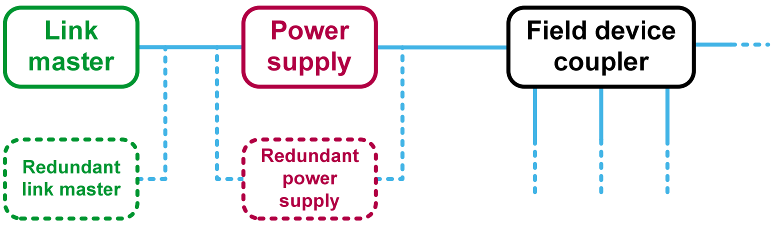


Figure 9: Redundant FOUNDATION Fieldbus power supply

The following figure shows several power supplies from different manufacturers providing redundancy capabilities:

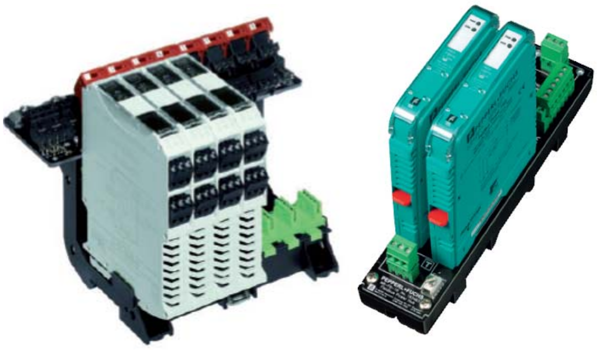


Figure 10: Redundant power supplies

The following figure shows several power supplies from different manufacturers without redundancy:

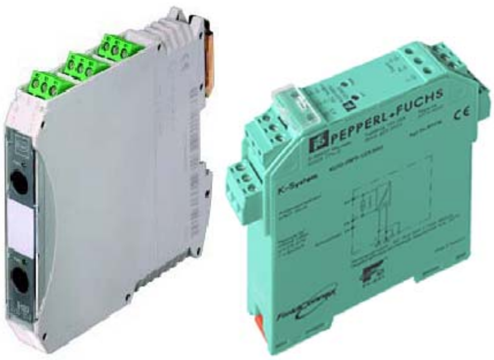


Figure 11: Non-redundant power supplies

We can conclude that the most important criteria when selecting a FF Power Supply are:

- Current
- Voltage
- IS requirements
- Redundancy

2.2.2. Cable

Several types of cable can be used in FOUNDATION Fieldbus networks. The following table shows the cable types defined in the physical layer standard (according to the FF-844 standard):

Type	Cable description	Size	Max. length
A	Shielded, twisted pair	#18 AWG (0.8 mm²)	1900 m
B	Multi-twisted pair with shield	#22 AWG (0.32 mm²)	1200 m
C	Multi-twisted pair, without shield	#18 AWG (0.13 mm²)	400 m
D	Multi-core, without twisted pairs and having an overall shield	#18 AWG (1.25 mm²)	200 m

Table 1: FOUNDATION Fieldbus cable types

Type A is the most common cable used. This kind of cable offers maximum distance and the best electromagnetic noise isolation. The picture below illustrates this type of cable:

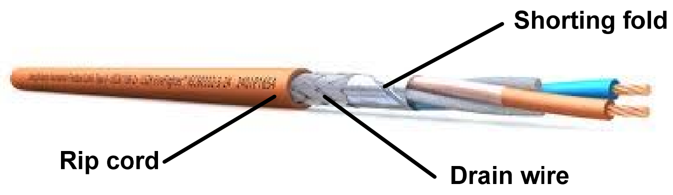


Figure 12: FOUNDATION Fieldbus cable – Type A

An alternative cable type that can be used is the multi-twisted pair (some manufacturers offer two to fifty pairs) with an overall shield, or type B. This cable type can be suitable when multiple buses exist in the same section of the installation, depending on the installation constraints (e.g. redundancy, availability and so on).

The other cable types (C and D) are usually not used because they offer a lower level of electromagnetic noise isolation and shorter distances.

The orange jacket is often used to identify the cables intended for Non-Intrinsically Safe areas and the blue jacket is used to identify the cables for Intrinsically Safe areas.

Belden, Northwire, Phoenix Contact and Rockbestos-Surprenant Cable Corporation (RSCC) were the first cable suppliers to pass the FOUNDATION Fieldbus registration process (source: FOUNDATION Fieldbus official website), although nowadays there is a wide range of manufacturers offering FOUNDATION Fieldbus cables. All certified FOUNDATION Fieldbus cables must feature the official checkmark.

2.2.3. Junction Boxes

Junction boxes can be considered the most common FOUNDATION Fieldbus cabling element. They act as a distributor to feed several FOUNDATION Fieldbus devices by means of trunk line derivation, commonly known as spurs.

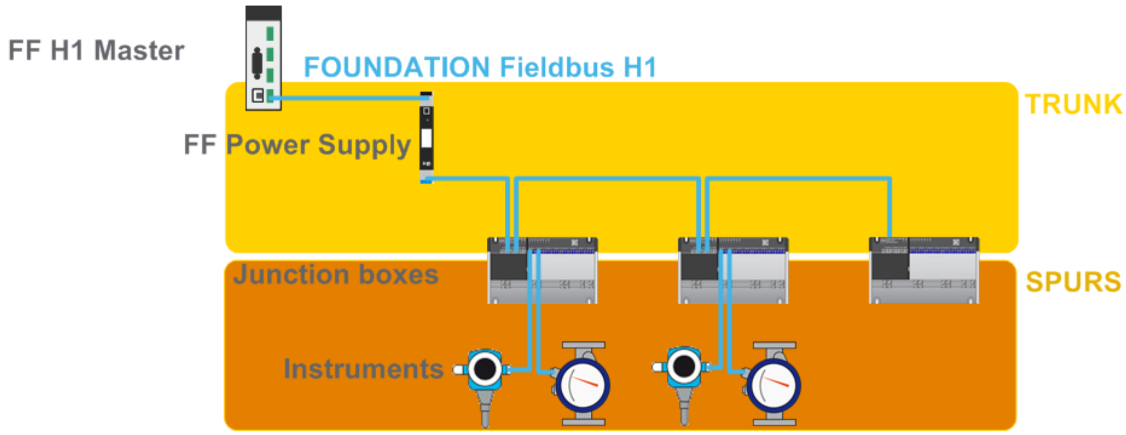


Figure 13: Architecture with STAHL junction boxes

Junction boxes exist in various designs, differing by:

- Number of channels (4, 6, 8, 12 and so on)
- Housing style (none, aluminum, stainless steel, polycarbonate, fiberglass and so on)
- Enclosure protection degree
- Temperature range
- FISCO/FNICO compliance

Figure 14: STAHL junction boxes

We can classify the junction boxes in two groups according to the electrical functionalities carried out by the junction box:

- The short-circuit protectors offer electrical protection against short-circuits produced on each spur. The trunk line is protected from short-circuits, preventing a problem in a device from shutting down the entire communication network. This kind of device is usually suitable for Non-Intrinsically Safe areas or Hazardous areas where the presence of flammable products is not constant (e.g. Zone 2 or 22 according to the ATEX directive).
- The galvanic isolators, in addition to the short-circuit protection, offer an electrical isolation between each spur and the trunk line. This type of device is mainly intended for installation in hazardous areas where the presence of the flammable products is occasional or constant (e.g. Zones 2, 22, 1 or 21).

⚠ DANGER

HAZARD OF ELECTRIC SHOCK, BURN OR EXPLOSION

Installation of devices in a hazardous area requires specific enclosures and installation considerations. Consult and apply the standards applicable in the geographic area where your plant is built and installed.

Failure to follow these instructions will result in death or serious injury.

2.2.4. Connectors

The FOUNDATION Fieldbus network does not require specific connectors because the bus components (such as power supplies, junction boxes and so on) commonly use systems to allow easy attachment to the bus. Therefore, installation includes usage of additional connectors to allow easy and fast modification of the bus components for maintenance.

The connector type commonly used is the screwed connector (available with different metrics). This type of connector usually offers an IP67 protection degree. Below is an example of this type of connector:

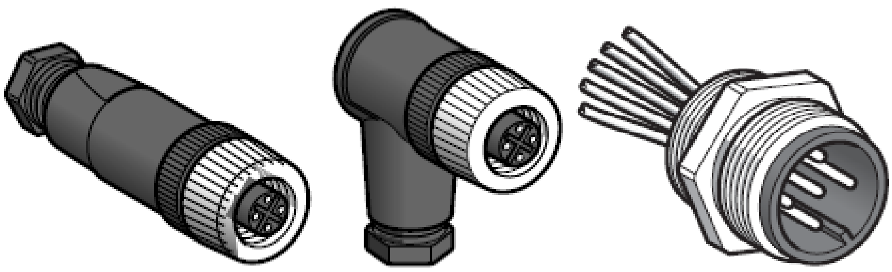


Figure 15: Cable connectors

An alternative method to create a spur is to use a connector commonly known as *T*. This connector provides a single derivation from the trunk line. This option is used less often due to its lack of electrical protection for the bus. Below is an example of a Phoenix Contact “T” connector:

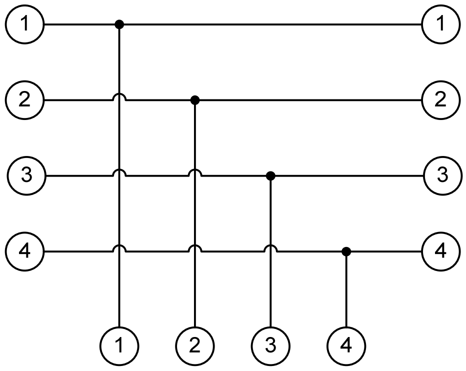


Figure 16: T connector

The installation constraints can require enclosures to be mounted. Several types of connectors and cable gland plugs (plastic or metal) can be present in enclosures. Below is an example of a junction box mounted in an enclosure with different connectors:

Figure 17: Enclosure with connectors

Some manufacturers offer pre-wired cables with connectors to reduce the overall installation costs and avoid common hard wiring errors, as shown below:

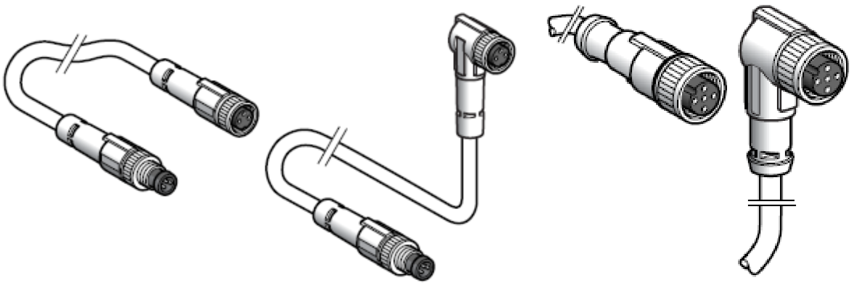


Figure 18: Pre-wired cables with connectors

The following picture shows several installation options using the pre-wired cables:

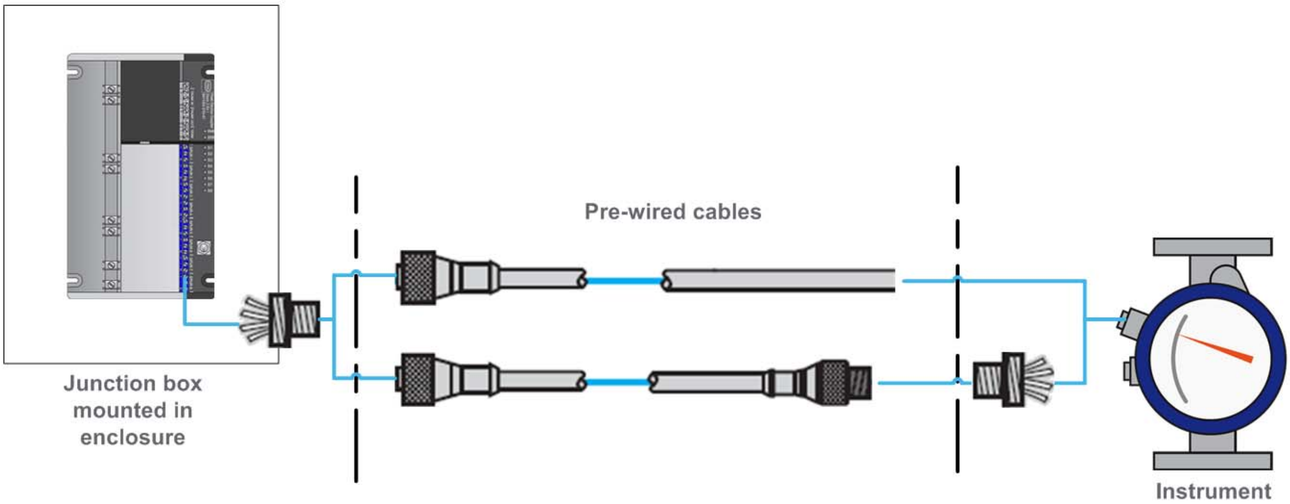


Figure 19: Installation options using pre-wired cables

⚠ DANGER
HAZARD OF ELECTRIC SHOCK, BURN OR EXPLOSION
Usage of connectors and cable gland plugs in hazardous areas must be compliant with the standards applicable in the geographic area where your plant is built and installed.
Failure to follow these instructions will result in death or serious injury.

2.2.5. Couplers

Specialized equipment is required to connect different signal types (such as digital I/O, temperature transmitters, valves and so on) which have not been designed with a FOUNDATION Fieldbus communication interface. Several manufacturers offer equipment to integrate these devices.

As an example, the following figure shows a digital I/O coupler from STAHL intended to connect Intrinsically Safe contacts, NAMUR actuators, valves or indicating lamps to FOUNDATION Fieldbus H1 segments – the FOUNDATION fieldbus communicator is included in the device.

Figure 20: Coupler

2.2.6. Terminators

A FOUNDATION Fieldbus H1 communication cabling also requires a specific type of bus terminator installed at each end of the cable segment in order to minimize signal reflections. This bus terminator is composed of one resistor (100 Ω) and one capacitor (1 µF).

Several possible forms are possible for a bus terminator, depending on the cabling type. This terminator can be installed as a termination connector, integrated in a tap unit, in a junction box by means of an integrated switch or it can be part of the FOUNDATION Fieldbus power supply.

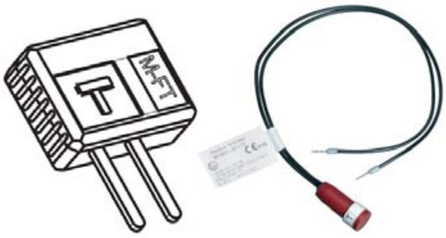


Figure 21: Cable terminators

2.2.7. Specific communications devices

Three extra types of devices can be connected to an H1 segment:

- Repeaters are intended to increase the distance of a single bus segment. The maximum number of repeaters in a segment is limited to four, so the maximum reachable distance is 9500 meters.
- Gateways are intended to provide connectivity towards other protocols like Modbus. Depending on the gateway, it can provide several physical medias (RS232 serial, RS485, Ethernet over copper and so on).
- Linking devices are intended to connect the H1 segments to the High Speed Ethernet (HSE) FOUNDATION Fieldbus networks. They pass the FOUNDATION Fieldbus protocol information between the different physical layers. This kind of device is also known as bridge.

All these devices are considered to be device participants in the segments and must be taken into account for the bus distance calculations and constraints.

2.2.8. Cable lengths and elements per segment

The cable type used in the installation limits the maximum reachable bus distance. The maximum distance of a H1 segment is 1900 m (using a type A cable).

The junction boxes introduce the trunk and spur concepts. The segment's total distance is calculated by adding the distance of both the trunk and the spurs. The following figure shows an example:

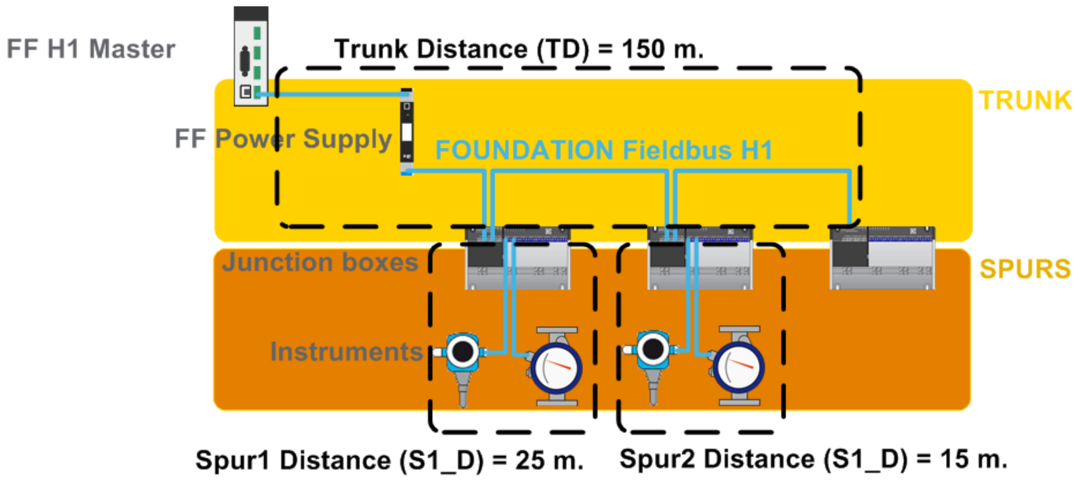


Figure 22: Trunk and spurs

The segment distance is calculated as the total of all the distances:

TOTAL Distance = TD + S1_D + S2_D = 150 m + 25 m + 15 m = 190 m

In addition, the spur length is restricted by the devices included on each spur and the number of devices in the segment. As an example, the following table shows the spur length recommendations for up to four devices (as recommended by the standards IEC-1158-2 and ISA S50.02-1992 Part 2, Annex C (informative) – these lengths are recommended but not required):

Devices	One device per spur	Two devices per spur	Three devices per spur	Four devices per spur
25-32	1 m	1 m	1 m	1 m
19-24	30 m	1 m	1 m	1 m
15-18	60 m	30 m	1 m	1 m
13-14	90 m	60 m	30 m	1 m
1-12	120 m	90 m	60 m	30 m

Table 2: Cable length recommendations (IEC-1158-2 and ISA S50.02-1992 Part 2, Annex C)

The maximum spur distance is 120 meters but it is recommended that the spur length be as low as possible.

Commonly the FOUNDATION Fieldbus device manufacturers provide software to check the viability of the installations. An example of this software will be shown later in this document.

2.3. Bus topologies

Several conceptual architectures for the H1 segments are described in this section. The distribution of the instruments on the bus is described here, but terminations and power supplies are not detailed. The architectures described below do not show bus redundancies, but it is possible to get two interfaces on the devices to have bus wiring redundancy.

2.3.1. Point-to-point

The point-to-point architecture is the simplest and perhaps the least used. It is possible in a segment containing only two devices.

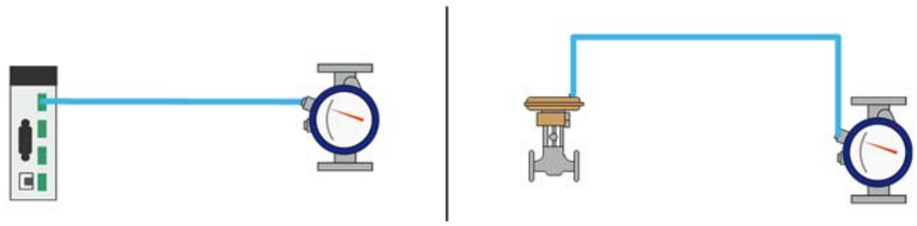


Figure 23: Point-to-point topologies

This architecture could consist of a transmitter connected to a control and monitoring system, or an isolated segment featuring a transmitter and an actuator in which one of them acts as a LAS

(Link Active Scheduler). This architecture does not offer a great advantage over the traditional wired systems and wastes the bus communication capabilities.

2.3.2. Bus with single spurs

The single spurs architecture is built by adding elements to the trunk line in order to get single spurs. These elements can be junction boxes or T connectors.

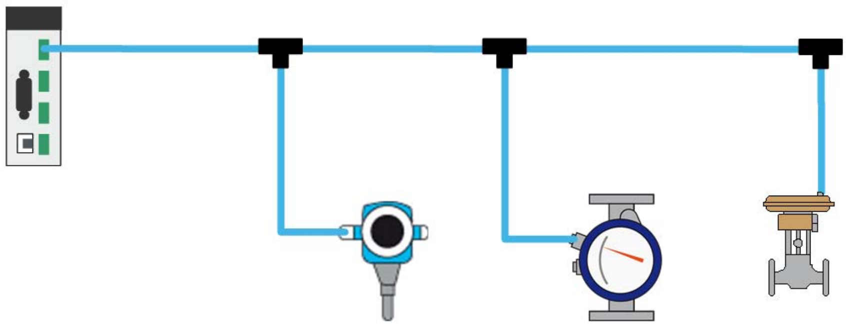


Figure 24: Single spurs topology

The length of the spurs, as discussed in previous chapters, may range up to 120 meters.

2.3.3. Daisy Chain

The daisy chain topology does not use derivations or spurs in the trunk line: no additional junction boxes or T connectors are used to build this topology. The trunk line is connected directly to the instrument terminals and extended to the next instrument through the same terminals.

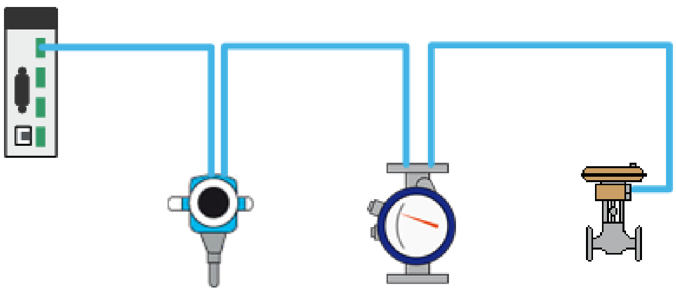


Figure 25: Daisy chain topology

Good practice when building this architecture is to use instrument connectors that allow the bus to be disconnected from the instruments without disturbing the communications or cutting the bus. This is useful in cases of maintenance or instruments replacement for example.

Note: With this architecture, no electrical protection is included to protect against short-circuits or instrument malfunction. Therefore, any problem in the bus will impact the entire segment.

2.3.4. Tree topology

In the tree topology, the instruments are connected to the trunk line using a single cable pair as spurs. The most common element used to build the spur is often the junction box. This kind of architecture is also known as chicken foot. An advantage of this topology is that the junction boxes usually include electrical protection, as discussed in the subsection 2.2.3. Therefore, an electrical issue on the spurs should not propagate to the trunk line.

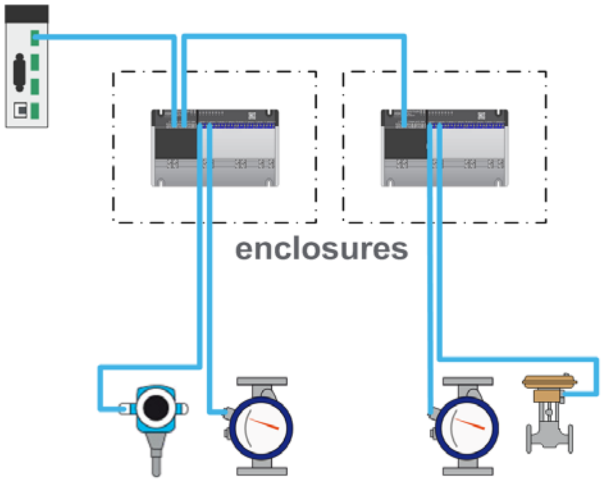


Figure 26: Tree topology

With this topology, the spur length must be carefully studied so as not to exceed the maximum length.

2.3.5. Combinations

It is possible to build a combination of all the topologies discussed above. As an example, the next figure shows a possible architecture:

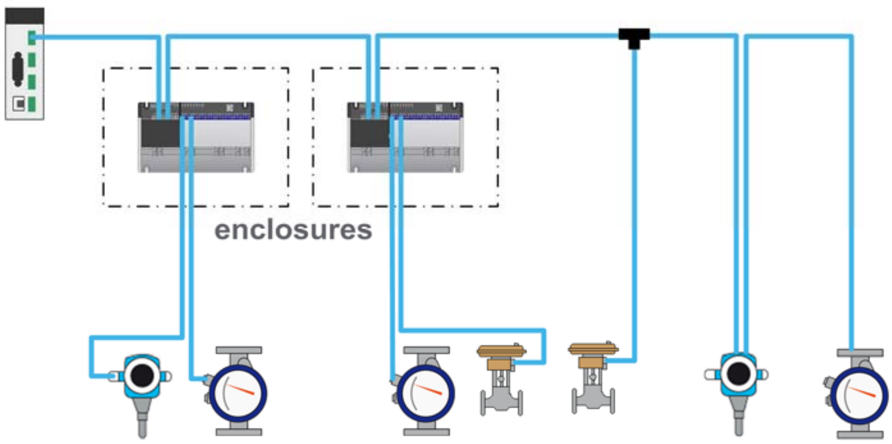


Figure 27: Combination of topologies

When mixing several architectures, the general rules about cabling distances, number of devices, and so on must still be adhered to.

2.4. Communication

The H1 FOUNDATION Fieldbus network is composed of several physical devices interconnected by a serial bus known as a link or segment. The fieldbus network comprises one or more segments and each segment has a unique identifier.

The physical devices which comprise the segment can be field devices (flow meters, temperature transmitters, actuators and so on) or host devices. For each physical device, an address, a device tag and a device ID are configured. The address and the device tag must be unique for each segment and should be configured by the user, while the device ID is configured by the device manufacturer – two devices should never have the same ID.

The FOUNDATION Fieldbus devices can perform process control functions. This is achieved when several mechanisms allow data exchanges between the devices and a bus manager, which performs the following actions:

- It ensures that the control loops execute with a scheduled timing
- It restricts more than one device from accessing the bus at the same time

To ensure these functions are performed, the H1 FOUNDATION Fieldbus uses a central communication system known as link master.

2.4.1. Blocks description

The FOUNDATION Fieldbus organizes the devices' memory in functional units called blocks. Organizing the memory like this allows the following:

- Interoperability: Devices can exchange information with each other independently from their manufacturer or brand
- Interchangeability: A device can be replaced with one from another manufacturer

To define these blocks the FOUNDATION Fieldbus specifies uniform device functions and application interfaces as well as data information structures. This allows other devices on the network to use these interfaces to exchange information.

The FOUNDATION Fieldbus assigns these functions and the device data information to three different block types:

- **Resource block**: The resource block specifies the general characteristics (hardware and software) of the device. That includes, for example, the device model designation and revision, manufacturer ID, serial number and resource state. It also offers some basic device control functions such as write protection, blocking operator access to the device and factory reset. It contains the state of all of the other blocks in the device. Each device has only one resource block.
- **Transducer block**: The transducer block operates as an interface between the physical sensor and the function block. The transducer block performs following functions:
    - Access the details of I/O devices
    - Signal linearization, filtering and calibration
    - Sensor measurement reading
    - Actuator driving

  There is one transducer block per sensor or actuator in the device – i.e. if the instrument has a flow measurement capability, there is a transducer block for this measurement.
- **Function block**: The function blocks determine the control and I/O behavior. Usually, a device has a set of functions it can perform. These functions are represented as function blocks within the device. During the installation, the user defines which block to use and the relationship between them via the monitoring and control application. Not all of the function blocks defined in the FOUNDATION Fieldbus standard are included on each device: the manufacturer chooses a set of function blocks to include on the device. The manufacturer can even include specific function blocks designed for the product. Alarm reporting and trending functionalities are also integrated in function blocks.

The following figure shows an example of a device's blocks distribution:

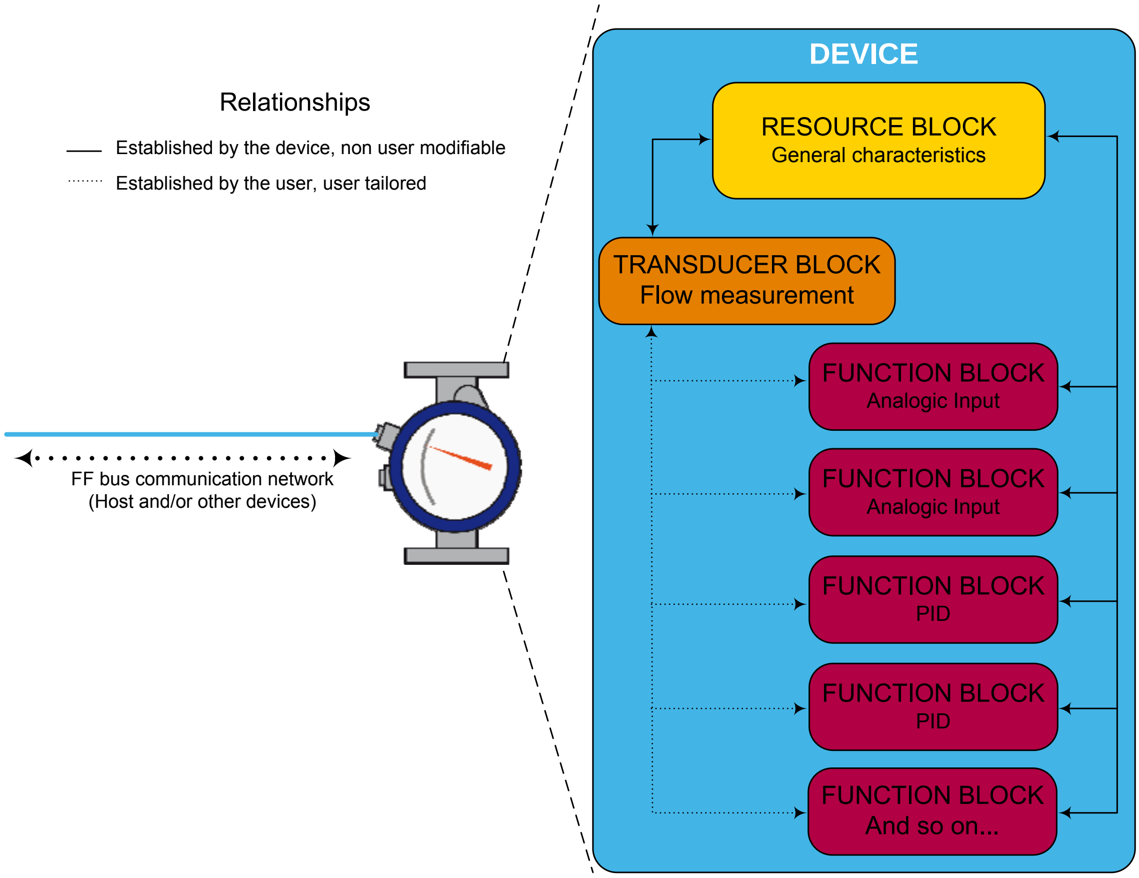


Figure 28: Blocks distribution example

Only one resource block is present per device. It contains information about the general characteristics of the device as well as the state information of the other blocks, thus being capable of providing diagnostics information. All these relationships are established by the device and can not be changed.

One transducer block is present for each sensor or actuator in the device. In the example above, the instrument can only perform a flow measurement. Therefore, there is only one transducer block.

The instrument manufacturer includes a set of function blocks which can fulfill the measurement functionalities the instrument is intended for. The user chooses which function blocks he wants to use during the instrument and system commissioning and can then create links between the function blocks and the transducer block.

The FOUNDATION Fieldbus function blocks can be classified depending of their usage level:

- Standard function blocks, defined in FF-891: Function Blocks – Part 2

Function block name	Symbol
Analog Input	AI
Analog Output	AO
Bias/Gain	BG
Control Selector	CS
Discrete Input	DI
Discrete Output	DO
Manual Loader	ML
Proportional/Derivative	PD
Proportional/Integral/Derivative	PID
Ratio	RA

Table 3: Standard function blocks

- Advanced function blocks, defined in FF-892: Function Blocks – Part 3

Function block name	Symbol
Pulse Input	–
Complex AO	–
Complex DO	–
Step Output PID	–
Device Control	DC
Setpoint Ramp Generator	SRG
Output Splitter	OS
Input Selector	IS
Signal Characterizer	SC
Deadtime	DT
Calculate	–
Lead Lag	LL
Arithmetic	AR
Integrator (totalizer)	IT
Timer	TMR
Analog Alarm	AAL
Discrete Alarm	–
Analog Human Interface	–
Discrete Human Interface	–

Table 4: Advanced function blocks

- Additional function blocks, defined in FF-893: Function Blocks – Part 4

Function block name	Symbol
Multiple Analog Input	MAI
Multiple Analog Output	MAO
Multiple Discrete Input	MDI
Multiple Discrete Output	MDO

Table 5: Additional function blocks

- Flexible function blocks, defined in FF-894: Function Blocks – Part 5. These function blocks allow the user or manufacturer to define block parameters and algorithms to design an application to interact with standard function blocks or host systems.
- Safety Instrumented function blocks, defined in FF-895.

Function block name	Symbol
SIF Analog Input	SIF-AI
SIF Discrete Output	SIF-DO

Table 6: SIF function blocks

Function block parameters

The function block parameters are classified as follows:

- Input parameters: Data received from another block
- Output parameters: Data sent to another block
- Internal parameters: Data contained within the block but not exchanged with other blocks

The parameters usually have multiple settings known as fields. For example, the parameter *OUT* is composed of two fields:

- *OUT.VALUE* contains the information to be exchanged
- *OUT.STATUS* indicates whether the value is good, uncertain or bad among other states

These types of parameters are generated by the block and published to the bus if it is required for other blocks in a separate device, otherwise they are not published.

The input and output parameters of a function block can be linked to another input/output function block parameter. The *STATUS* information is contained either in an input parameter or in an output parameter.

The internal parameters cannot be linked to an input/output parameter and do not contain *STATUS* information. They are preconfigured by the device manufacturer, even though some of these parameters should be set by the user during the commissioning. An example of this kind of parameter is the scaling of a measurement.

Function block application

Specific software tools (e.g. NI-FBUS or FF-CONF) create function block applications, i.e. configure the main parameters of the function blocks and possibly create links between the function blocks. Once this application is generated, it should be downloaded to the FOUNDATION Fieldbus network. Each device uses its local function blocks application and the corresponding main parameters (specific device parameters are configured with their DTMs).

Function block MODE_BLK parameter

Each function block has a mechanism to control the block execution, independently of the application downloaded in the device. This functionality is configured by an internal block parameter called *MODE_BLK*. It is possible to control the block execution through this parameter. The most common execution modes are:

- Automatic (*AUTO*): The block uses a local setpoint value received by the transducer block to determine the output value
- Cascade (*Cas*): The block receives its setpoint value from another function block to determine the output value
- Manual (*Man*): The operator writes the output of the block directly
- Out Of Service (*OOS*): The block is not running – this mode is usually used during block configuration. Some devices require that the function block is in *OOS* mode when changing certain parameters

The *MODE_BLK* parameter comprises four fields, which are based on the previously presented modes:

- *PERMITTED*: This field is not user configurable – it contains a list of the allowed modes for the particular block
- *ACTUAL*: This field is not user configurable – it shows the current operating mode of the particular block
- *NORMAL*: This field is user configurable – it sets the mode that the block should run during normal operating conditions
- *TARGET*: This field is user configurable – when writing this field, the operator requests a specific execution mode for the particular block. The device will try to change the execution mode and if it is successful the *ACTUAL* field changes and shows the same value as the *TARGET*

Objects

In addition to the blocks described above, the user layer of a FOUNDATION Fieldbus device contains other object types. The objects are defined by the FOUNDATION Fieldbus specification as part of the function block application structure. Usually, the FOUNDATION Fieldbus configuration software package manages a transparent way to deal with these objects so that the user should only set up the behavior and relationships between the blocks. Below is a list of some objects:

- Linkage object: Defines the connections between the outputs of one block and inputs of another, whether the blocks are in the same device or different devices.
- View object: Allows efficient transfer of groups of parameters. This provides easy access to the parameters for supervision applications.
- Alert object: Allows a device to report alarms and events to the fieldbus. The alert objects are fully configurable by the user.
- Trend object: Accumulates values of function block parameters for access over the network and publishes historical data for supervision trending and storage. They include the parameter value, status and a timestamp. Multiple parameters in the same block can be trended. Other devices or hosts over the network could use the information accumulated by the trend object.

2.4.2. Electronic Device Description (EDD)

The Electronic Device Description (EDD) is a set of files which contains a description of the data and functionalities contained in a device. For the most common functionalities (such as function and transducer blocks), the FOUNDATION Fieldbus organization has available device descriptions, even though the manufacturer can add specific information or functions by defining these parameters on the device description.

One device description exists for each device type. It consists of files with the extensions *.ffo*, *.sym* and *.cff*. The file used by the FOUNDATION Fieldbus configuration software tools to import a new device is the *.cff* (common file format), which is encoded with the ASCII file format. This file is written using the Device Description Language (DDL), which is then treated with the help of a tool called Tokenizer to generate the final EDD. The control systems will use this file to know the functionalities available in the device in order to implement the control/device network.

The main advantages of the EDD are:

- Computer operating system independence
- Control system platform independence
- Interoperability with different devices and hosts manufacturers

2.4.3. Link master and LAS

The link master is a device that can manage the communications traffic on a segment by broadcasting different commands to the devices. For each FOUNDATION Fieldbus network, at least one link master should be present.

Several devices with link master capabilities can be present on the network at one time but only one of them will control the communication – the chosen device is activated during the network configuration. The link master controlling the bus is known as the LAS.

The LAS should be able to:

- Recognize and add new devices to the controlled segments
- Remove non-responsive devices from the list of active devices
- Synchronize all device clocks on the controlled segments
- Poll the devices
- Manage the communication priorities of the segments (token ring)

The LAS uses scheduled and unscheduled data transmissions and the importance of the data determines which one is used. The time critical tasks, such as control loop, use the scheduled services while the configuration, diagnostics, alarms and events use the unscheduled services.

Scheduled communication

Scheduled communications are intended to avoid access conflicts and all corresponding tasks are executed cyclically in a strict scheduled order, which is defined during the network configuration.

A very important function accomplished by the LAS is the time synchronization of all the devices on the network. The LAS cyclically broadcasts a synchronization signal called Time Distribution (TD) so that all the devices on the segment have the same link time. This functionality is automatically managed by the system.

To illustrate how the LAS' scheduled communications work, see the example of a H1 segment block scheduling below:

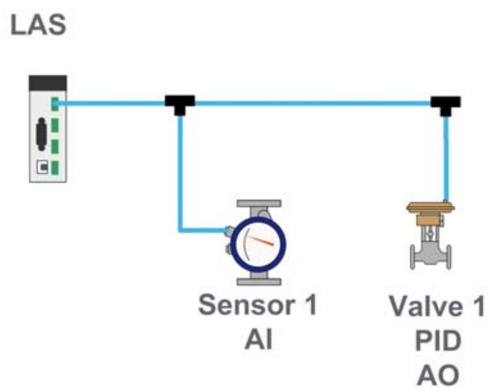


Figure 29: Simple H1 segment architecture

The system includes the following elements:

- One LAS
- One measurement instrument
    - Analog input function block
- One actuator (valve)
    - PID function block
    - Analog output function block

The system can be represented as follows:

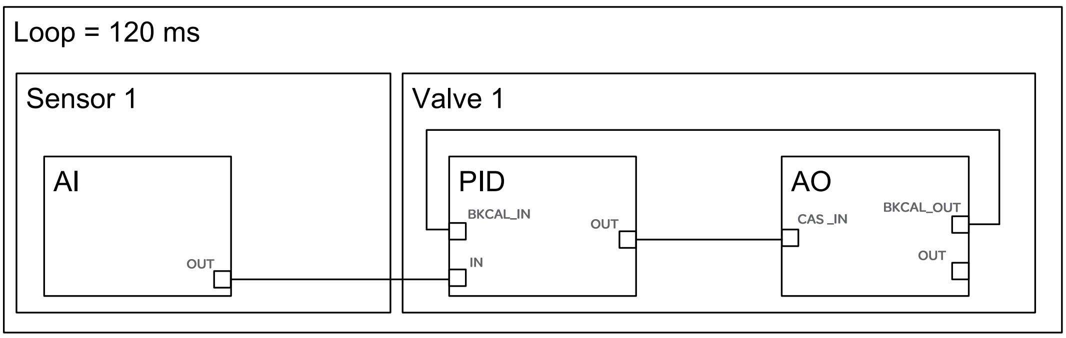


Figure 30: Simple H1 segment function blocks

The schedule system determines which activities (i.e. AI, AO, PID and so on) are running on the devices at a given time and, consequently, when data is sent to the network. Each activity is scheduled for a certain amount of time (macro cycle) and planned to start at a specific time (offset to the beginning of the complete cycle). The schedule generates a transmission list which defines when a specific device should send its data to the network. This list is automatically created by the bus configuration tool (e.g. NI-FBUS).

The following table shows the transmission list associated to our example:

Task	Description	Actions	Time-offset
A	Sensor 1	Execution - AI	0
		Data transmission – AI	20
B	Valve 1	Execution – PID	30
		Execution – AO	45

Table 7: Simple H1 segment transmission list

This schedule is sent to the network and the LAS stores the transmission list while each device stores only the information concerning its own behavior. All the devices on the bus are synchronized – the reference time is common to all devices and considered as the beginning of the schedule.

Each time a device is due to perform a data transmission, the LAS sends a command called Compel Data (CD) to the device. When the message is received by the device, the device (publisher) broadcasts the data in the buffer to all devices which are configured to receive the data (subscriber). The FOUNDATION Fieldbus uses the publisher/subscriber mechanism.

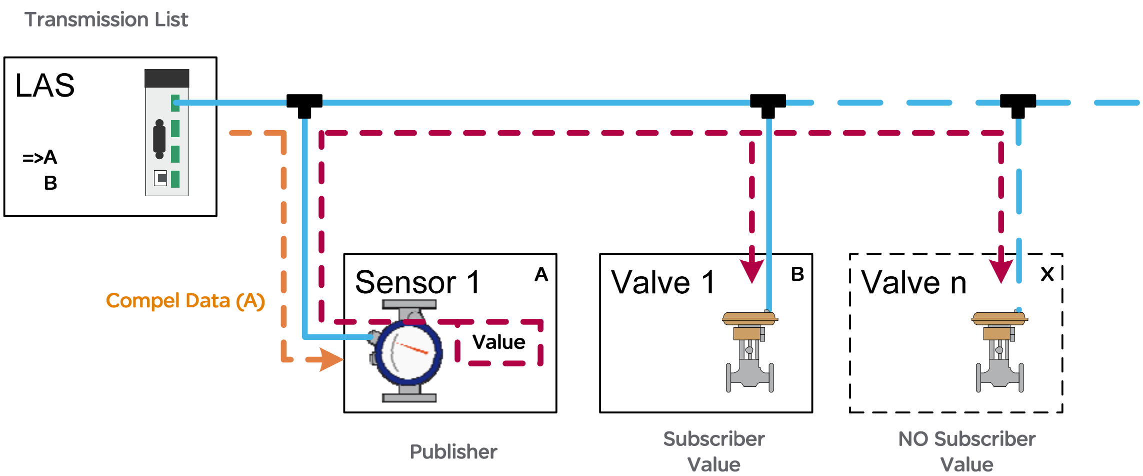


Figure 31: Data flow on the architecture

The message is published for the entire network and devices, but only the subscribers (e.g. the valve1 in our example) take the value from the network and treat the value according to their programming.

The following time chart and table illustrates the example discussed above:

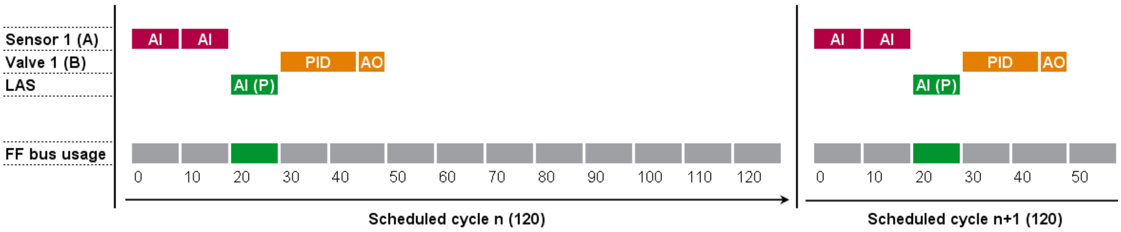


Figure 32: LAS time chart

Time	Action description
N+0	• Sensor 1 starts the measurement
N+20	• Sensor 1 should be finished with the measurement, so the LAS sends the *CD* command to Sensor 1 • Once Sensor 1 receives the *CD* command, it publishes the measurement value on the bus. • At this moment, the Valve 1 is ready to receive the value.
N+30	• Valve 1 starts executing the PID
N+45	• Valve 1 has finished executing the PID • Valve 1 executes the AO with the current PID output value.
N+120	• The scheduled cycle starts again

Table 8: LAS time table

As you can see in the scheduled cycle, the bus is only used when necessary.

Unscheduled communication

Unscheduled communications are intended for non-time-critical information related directly to the process – for example, device configuration, diagnostics, event notification alarms and trend reports. This type of communication is only allowed when the bus is not being used for the scheduled communication. This is achieved using the pass token mechanism.

The LAS manages the permissions for each device. The LAS sends the pass token (*PT* command) to all devices and each is allowed to use the bus until the token returns or the maximum time allowed for a device is reached.

The LAS needs to get actualized information about the bus-connected devices. For this purpose, it uses a table called *live list*, which is continuously updated by sending the Probe Node (*PN*) command. If a device responds with the special command Probe Response (*PR*), it is added to the live list, while the device that does not respond to the *PR* command is removed from the list.

The devices are also removed from the list if there is no Token response after three consecutive tries.

In the example used for the scheduled communication, the time ranges available for the unscheduled communication are from N to N+20 and from N+30 to N+120.

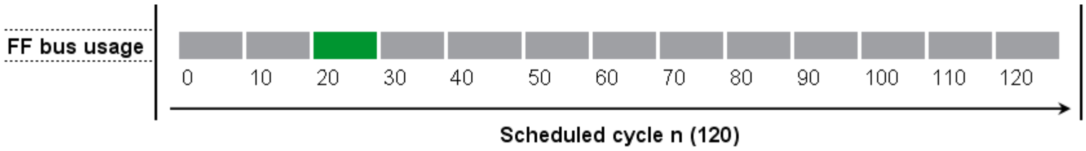


Figure 33: Communication bus usage

3. Selection

This chapter details the hardware and software choices made for the example project referenced in this document.

3.1. Hardware

Only devices with the FOUNDATION Fieldbus certification should be connected to the FOUNDATION Fieldbus network.

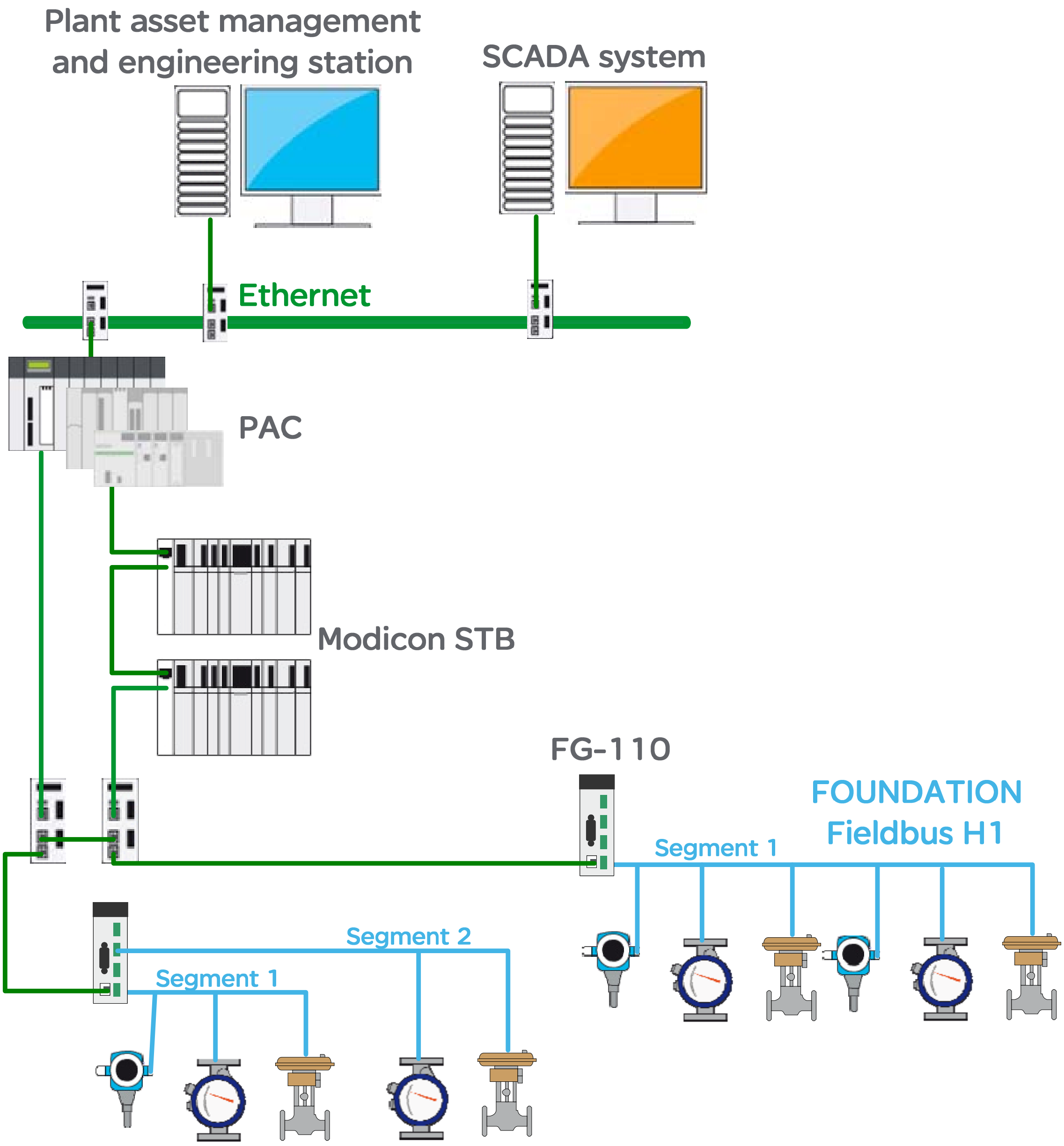


Figure 34: Architecture example

This document demonstrates how to connect a PlantStruxure PAC and SCADA to the H1 FOUNDATION Fieldbus network and shows the following related technical means:

- A gateway from our partner Softing is selected to provide connectivity between Modbus TCP and FOUNDATION Fieldbus networks
- The selected PAC is a Quantum standalone controller – the detailed descriptions provided for the Quantum in this document can generally be applied to other Unity Pro based controllers (M340 and Premium)
- Schneider Electric ConneXium switches are used for the physical connection between the PAC, the gateway and the SCADA system using Modbus TCP/IP as a backbone protocol
- FOUNDATION Fieldbus instruments are connected to the H1 network

3.1.1. Softing FG-110 FOUNDATION Fieldbus linking device

Softing designed the FG-110 FF device that covers the functionalities of:

- The gateway: This device acts a gateway between the FOUNDATION Fieldbus and Modbus protocols
- The linking device: This device allows the H1 segments to connect to the HSE networks. The linking device provides interoperability between the physical mediums

The gateway does not provide power to the FOUNDATION Fieldbus H1 segments so an external FOUNDATION Fieldbus certified power supply must be added for each segment.

Figure 35: FG-110 Linking Device

The FG-110 FF allows up to 64 FOUNDATION Fieldbus H1 devices to be distributed in four H1 segments. Each segment can contain up to 16 devices.

The FG-110 FF has two Modbus ports:

- Serial Port RS232 through a DB9 connector
- Ethernet Port through a RJ45 connector

The FG-110 delivers a wide range of function blocks and parameters to allow control in a field application using Modbus.

Note that the H1 link redundancy is possible by using two FG-110 linking devices. This topic is not discussed in this document; for further information about this feature, please refer to the FG-110 Softing application notes.

The FG-110 FF can be used as a key component for implementing additional automation functionality in the FOUNDATION Fieldbus section of the plant. For instance, asset management

systems and FDT frame applications (e.g. M&M, FieldMate, Field Device Manager or PACTware) can be included using the FG-110 as an access point. Unity Pro includes an embedded FDT container, and has done since version V5.0. An example of the UNITY FDT container is described later in this document.

3.1.2. FOUNDATION Fieldbus power supply

To make sure the devices on the segment have enough voltage, a minimum of 9 V is required. The following information is also necessary:

- Current consumption of each device
- Physical location on the network of each device
- Power supply physical location on the network
- Cable section resistance
- Power supply voltage

All these physical requirements can be checked using standard tools such as DesignMATE (available on the FOUNDATION Fieldbus web site for free).

For the project built for this document, we have chosen a power supply from STAHL – reference 9142/00-310-11s.

The FOUNDATION Fieldbus power supply is only intended for the supply of energy to a FOUNDATION Fieldbus H1 segment, i.e. field devices and the host.

This power supply model allows a segment to be supplied with power redundantly or with increased output current (boost operation) by adding a second power supply.

Figure 36: STAHL FF H1 power supply

Each power supply module has an integrated switch to activate the bus termination for standard-compliant segment end termination.

The Fieldbus Power Supply is connected to the host using the trunk line, delivering a current up to 540 mA in boost mode.

For diagnostic purposes, the power supply has a contact relay in case of overload or a short-circuit.

3.1.3. Junction box

The junction box used in this project is the ISbus series, reference 9411/24 from STAHL.

Figure 37: STAHL junction box

This junction box is used to connect up to eight field devices. It works on a physical level, i.e. independently from the protocol used, as long as it complies with the IEC/EN 61158-2 standard. A terminating resistor is embedded and can be activated or deactivated with a jumper.

This junction box can be installed in hazardous areas. For further information, please refer to the device manual.

This junction box is equipped with galvanic isolation between the trunk and spurs, which provides short circuit protection for each spur with a functional current limitation of 50 mA.

For diagnostics purposes, the device is equipped with several LEDs to indicate the trunk voltage and spurs status.

3.1.4. Instrumentation

The instruments selected to build the project described in this document are:

Manufacturer	Reference	Instrument type	Available function blocks
KROHNE	H250 M40 ESK4	Variable area flow meter	• Two analog inputs (AI) • One integrator (IT) • One proportional integrate derivative (PID)
Endress+Hauser	TMT85	Temperature sensor	• Three analog inputs (AI) • One input selector (IS) • One proportional integrate derivative (PID)
SAMSON	3730-5	Electropneumatic positioner	• Two digital outputs (DO) • One input selector (IS) • One multiple analog input (MAI) • One multiple analog output (MAO)

Table 9: instruments summary and FOUNDATION Fieldbus function blocks features

The selected instruments are shown below:

Figure 38: Instruments used for this project

Note: KROHNE is a member of Schneider Electric CAPP (Collaborative Automation Partner Program).

3.1.5. PAC

No specific Modbus feature is necessary to integrate the FG-110 linking device with the PACs. Any Unity Pro based PAC can be used. We selected a Quantum PAC for this document.

The PAC must drive all the data exchanges with the FG-110 linking device, which will act as a server in the communication schema. In our application example we use the I/O Scanning services of the Quantum NOC DIO module. This module provides the interfaces to communicate with the connected devices using the DTM technology. Furthermore, it provides transparency to the equipment connected to the control network (e.g. SCADA and asset management tools) down to the devices through the Softing gateway. This allows instrument commissioning or connection to the network devices' different web servers.

For this project, we use the Quantum CPU 651 50 in combination with the Quantum NOC 780 00 module – this architecture can also be implemented with M340 or Premium PACs.

3.1.6. Selected network topology

The selected architecture for the H1 segment is the tree topology in a single segment using a non-redundant FOUNDATION Fieldbus power supply. The instruments selected for this project are connected to the same STAHL junction box. The cables used to build the H1 segment are Turk type A cables.

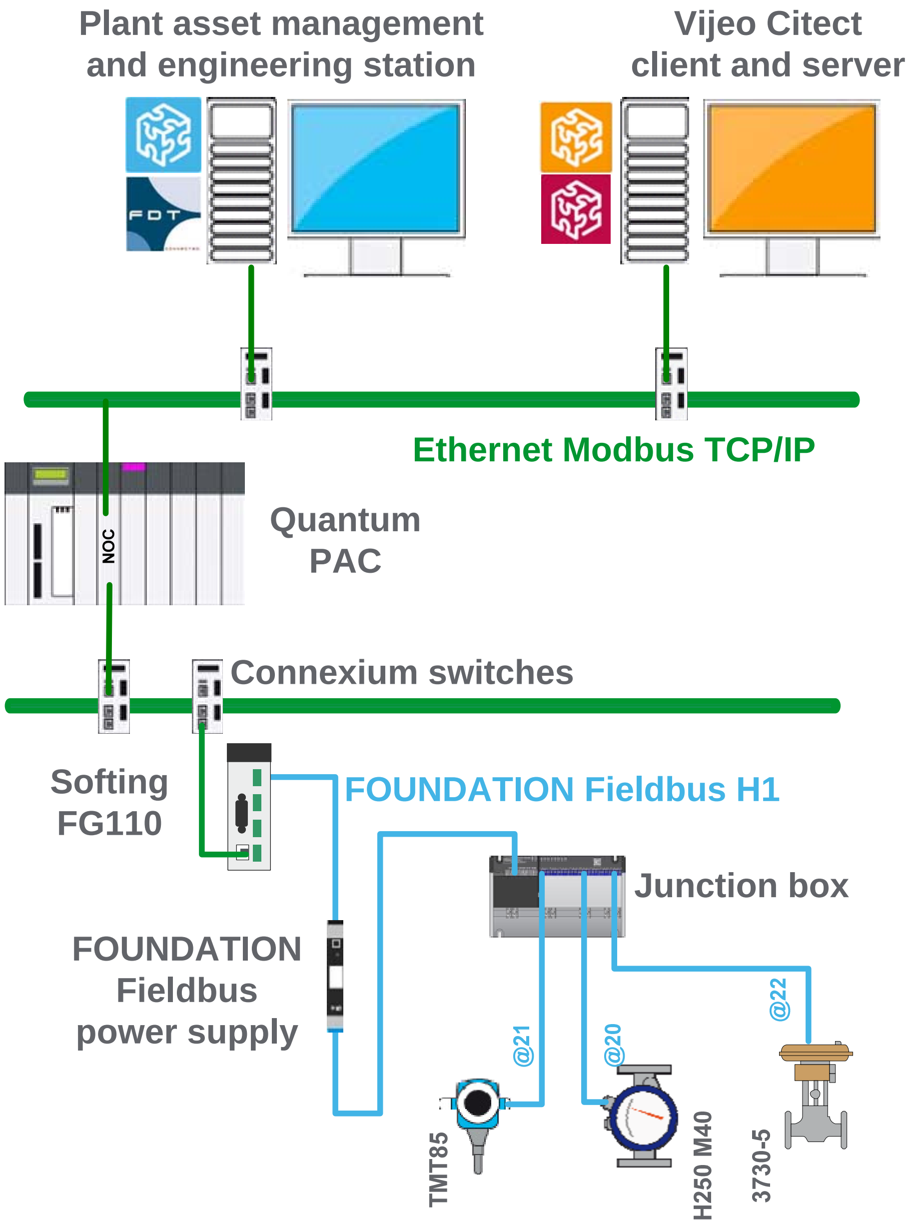


Figure 39: Architecture used for this document

The PAC is connected to the system using the NOC card, as depicted below:

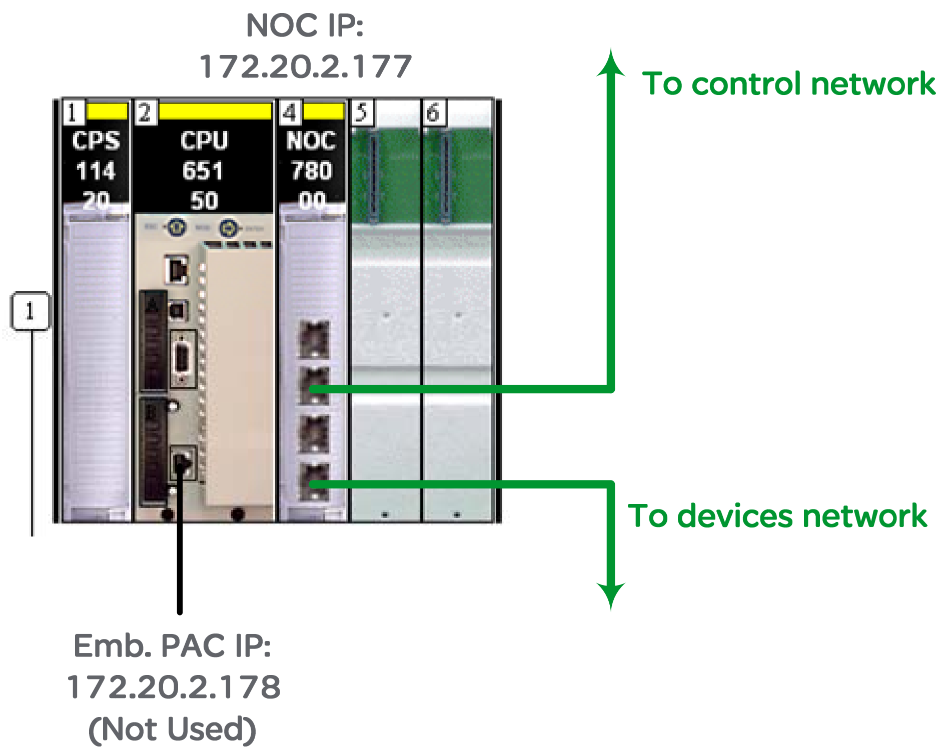


Figure 40: PAC connection

The following table presents the Ethernet network configuration of the devices, i.e. the IP addresses and the mask for the different devices' ports:

Device	Port	IP Address	Mask
SCADA server and client	PC	172.20.2.184	255.255.0.0
Asset management tools	PC	172.20.2.185	255.255.0.0
PAC	Embedded	172.20.2.178	255.255.0.0
NOC	Service	172.20.2.177	255.255.0.0
	Interlink		
	Dev. Network 1		
	Dev. Network 2		
FG-110	Modbus TCP	172.20.2.175	255.255.0.0

Table 10: Ethernet network configuration

3.2. Software

This section describes the software selected for this project and the minimum recommended system requirements. For detailed information about the hardware and software requirements, please refer to the specific product manuals.

3.2.1. FG-110 embedded web server

The FG-110 FF linking device's embedded web server offers the following functions:

- Configuration of the linking device (e.g. IP address settings or Modbus mapping)
- Provision of diagnostic information on the FOUNDATION Fieldbus or Modbus as well as monitoring of process values of the connected field devices

3.2.2. Softing FF-CONF

The FF-CONF is a Windows-based configuration tool which provides functionality to define all the required FOUNDATION Fieldbus network and device settings. This includes the following:

- Definition of function block linking and scheduling
- Bus parameter settings
- Field device parameterization

Figure 41: FF-CONF

Once the configuration has been completed in offline mode, the configuration data is downloaded to the network and field devices via the Softing hardware. The connected field devices are displayed in the Live List. In addition, the online mode supports read and write access to the device parameters and allows process values to be monitored. For individual configuration tasks, the FF-CONF includes a graphical user interface, which is designed for offline configuration and Live List display.

3.2.3. Unity Pro

Unity Pro XL is used to implement the code embedded in the selected Quantum PAC.

3.2.4. OFS

Schneider Electric's OPC Factory Server (OFS) software has been selected so that the SCADA system Vijeo Citect can access the data of the Schneider PACs.

3.2.5. Vijeo Citect

Vijeo Citect is the operating and monitoring component of PlantStruxure. With its powerful visualization capabilities and operational features, it delivers actionable insight faster, enabling operators to respond quickly to process disturbances and thereby increase their effectiveness.

4. Design

This chapter describes how the architecture and application are designed for the example detailed in this document.

4.1. Dimensioning the FOUNDATION Fieldbus gateway

Four H1 segments can be managed by the FG-110, with up to 16 devices on each segment.

Note: The number of devices could be lower in hazardous areas due to installation constraints.

On the Modbus side, the following information is available:

- FOUNDATION Fieldbus function blocks: AI, MAI, AO, MAO, DI, DO, PID, IS and OS
- Function blocks parameters: Actual Mode, Mode Target, IN, SP, OUT, PV, Channel, Block_Error and Status

The current firmware version of the gateway accepts up to 15 client/server Modbus TCP/IP connections. The available Modbus mapping starts at the address 40001 and the memory zone 49xxx is reserved to the gateway statistics information. The I/O Scanning configuration should be adapted to these limits.

4.2. FOUNDATION Fieldbus design software

The FOUNDATION Fieldbus organization provides the software design tool DesignMATE (available for free on the FOUNDATION Fieldbus website) to help design and check the H1 segments at the beginning of a project. DesignMATE checks the system consistency depending on the cable length, number and type of devices, power supplies and so on.

The figure below shows an example of this tool, building our architecture example:

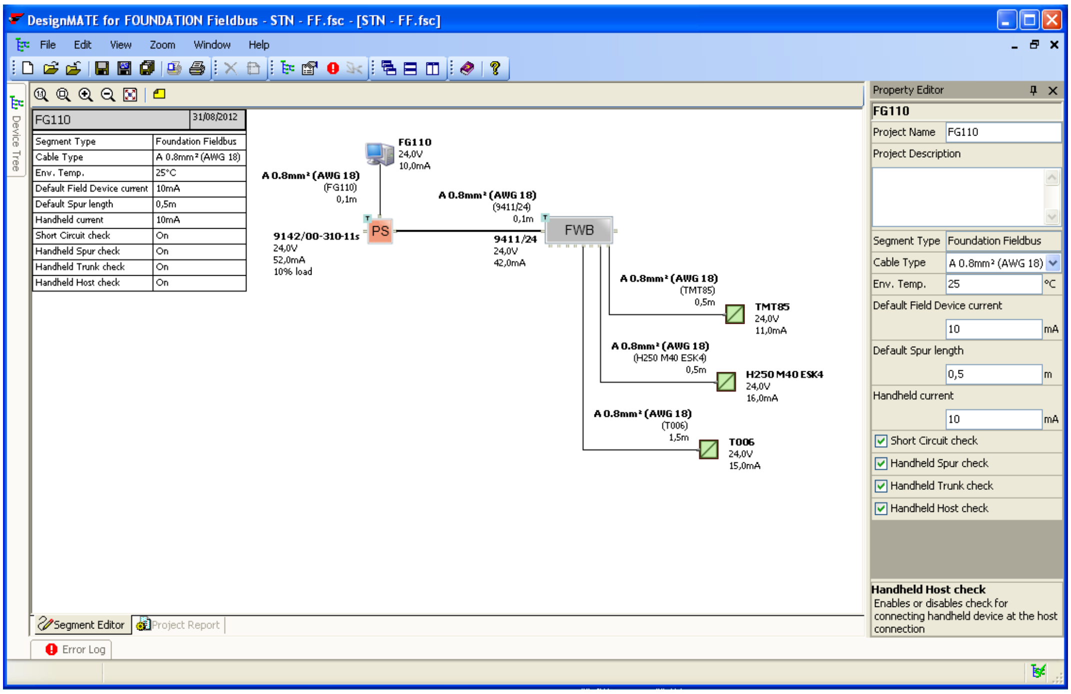
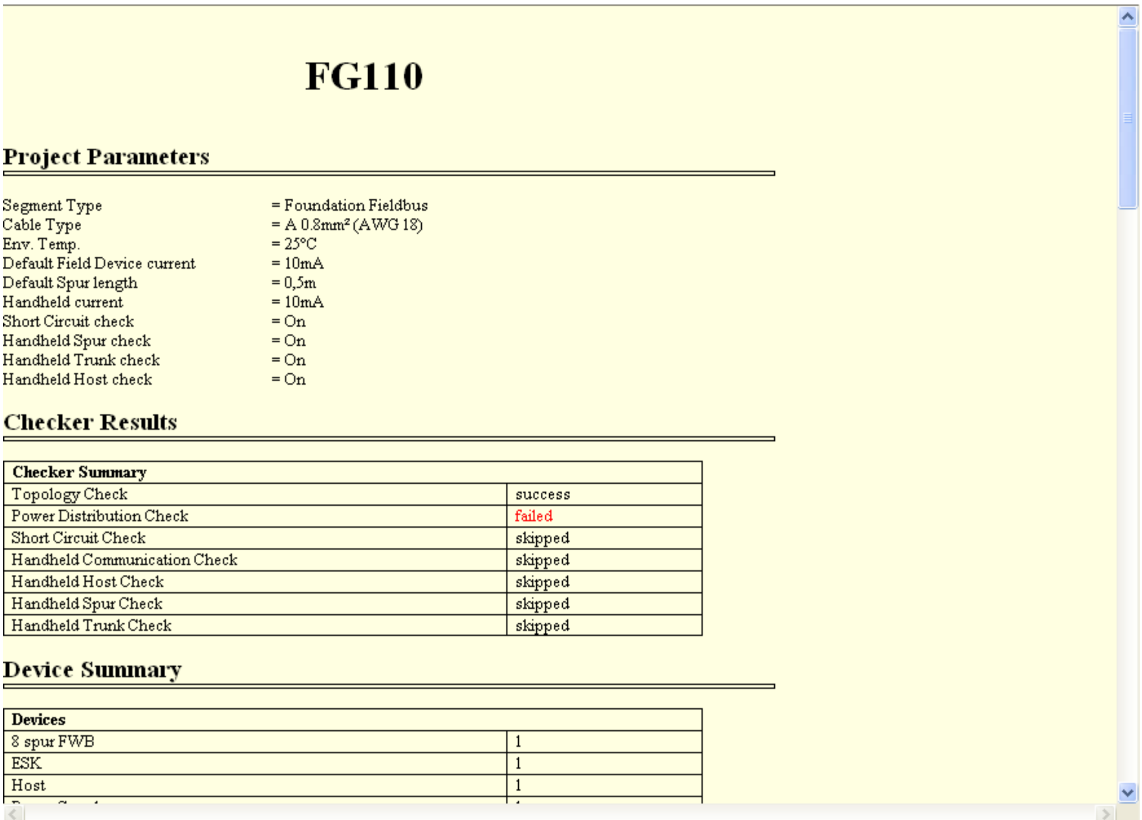


Figure 42: DesignMATE – example system view

After completing the layout editing, you may view the project report screen, which shows a project summary, followed by specific details for each network element:



FG110

Project Parameters

Segment Type	= Foundation Fieldbus
Cable Type	= A 0.8mm² (AWG 18)
Env. Temp.	= 25°C
Default Field Device current	= 10mA
Default Spur length	= 0,5m
Handheld current	= 10mA
Short Circuit check	= On
Handheld Spur check	= On
Handheld Trunk check	= On
Handheld Host check	= On

Checker Results

Checker Summary	
Topology Check	success
Power Distribution Check	failed
Short Circuit Check	skipped
Handheld Communication Check	skipped
Handheld Host Check	skipped
Handheld Spur Check	skipped
Handheld Trunk Check	skipped

Device Summary

Devices	
8 spur FWB	1
ESK	1
Host	1

Figure 43: DesignMATE – project report example

The project summary reports the result of a topology check (which could detect a missing device) and the result of a power distribution check (which detects whether the overall device power requirement exceeds the power supply capacity). Furthermore, it provides detailed information about the estimated consumption of each device, device summary, connections list and so on.

4.3. FOUNDATION Fieldbus data exchange and process scheduling

The data exchange configured for this project is simple in order to focus on the required steps to configure the system. Most of these steps must still be followed for more complex communication.

The following devices are available: a temperature sensor, a flow meter and a valve with electropneumatic positioner. We configure a PID to control the flow in the pipe using the FOUNDATION Fieldbus PID block of the H250 flow meter.

The AI block is used to read the value of the H250 flow meter, which delivers the measurement to the PID block (also executed on the H250 flow meter). The PID block is connected to the AO block, which controls the valve actuator, and the specific output of this block *BKCAL_OUT* is used as a feedback value to the PID to get the deviation from the desired value.

An additional AI block is added in the TMT85 to get the temperature measurement. This information is not linked to any other FOUNDATION Fieldbus block but the information delivered by the AI block is forwarded to the Modbus control system. This information is scheduled in the LAS (FG-110) in a macro cycle executed every second – this is the default value, which you can change, but you have to make sure that the LAS can handle the data flow on the bus according to your system.

The following figure shows the FOUNDATION Fieldbus blocks and their connections, representing the H1 FOUNDATION Fieldbus control application:

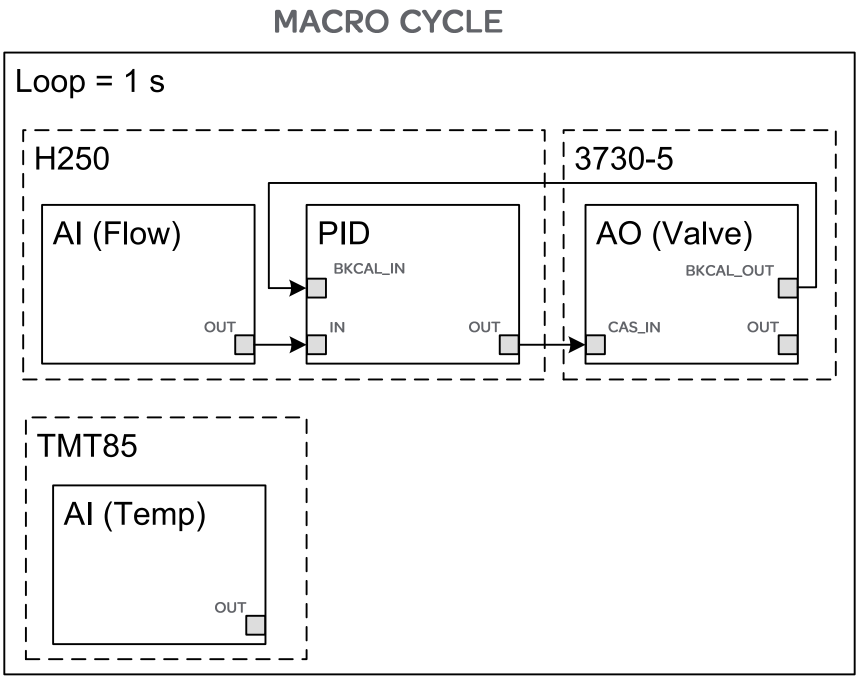


Figure 44: Function Block connections

The information exchanged with the Modbus control system is detailed below:

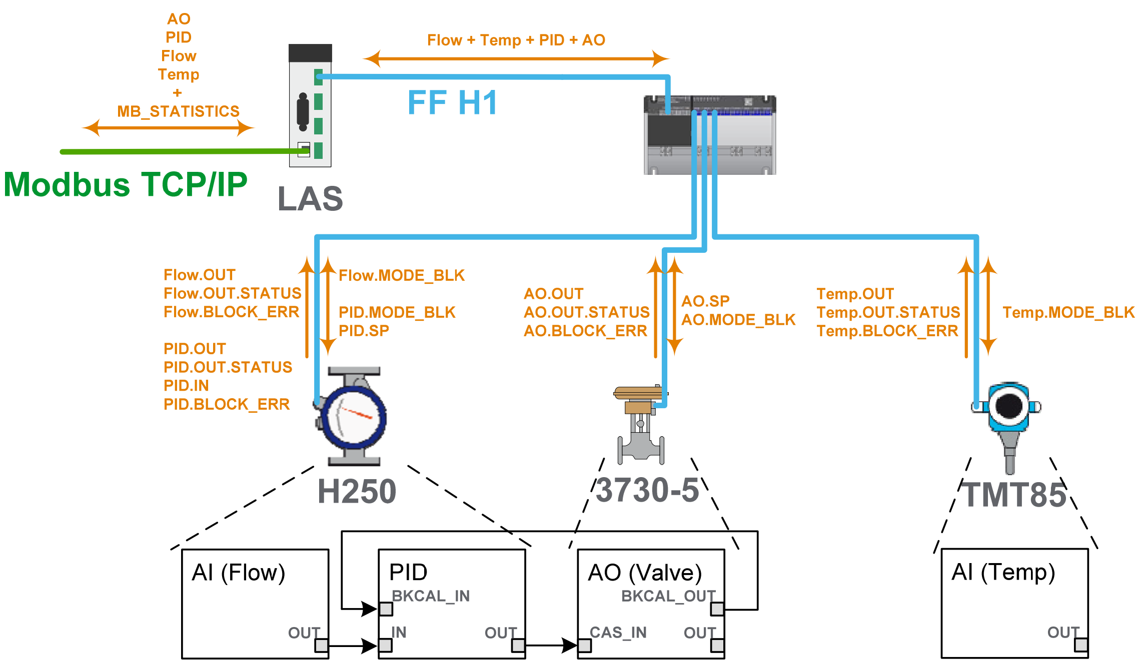


Figure 45: Data exchange using Modbus TCP

One AI block is declared in each instrument to retrieve the output value (measurement) and the output value STATUS. Furthermore, all the available statistics information is sent to the FG-110 linking device.

4.4. Software design

Following the PlantStruxure philosophy, the application will meet the next recommendations:

- Modeling of the functionalities used in the system, through DFBs on the PAC and Genies on the SCADA system
- Structuring of the system data

Applying these recommendations leads to the following benefits:

- Standardization
- Reliability
- Reutilization
- Lower engineering time

4.4.1. PAC

A DFB is associated to each function, along with the corresponding DDTs (Derived Data Type). In the examples below, several functions are presented – some are part of Schneider Electric's Device and Process Libraries (DPL) for PlantStruxure, while some of them have been specifically developed for this project.

Device and Process Library DFBs	Specifically developed DFBs
Analog input Analog alarm Digital input Multiple analog input Condition summary Control valve with position feedback	Processing measurement FG-110 Modbus Statistics

Table 11: DFBs list

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION Use all the diagnostics at your disposal (instrument status – may be configured using the DTM of the instrument – fieldbus status and gateway health status) to make sure the data you are using in your application is valid. **Failure to follow these instructions can cause death, serious injury or equipment damage.**

Measurement processing

The DFB *FF_MEASURE* was specifically developed to group the measurement value and the corresponding status information with the treatment of the status byte. Thanks to the NOC module, we can directly get the measurement mapped on Modbus in a real format. However, the status byte associated to the measurement has to be decoded to get accurate information about the measurement value.

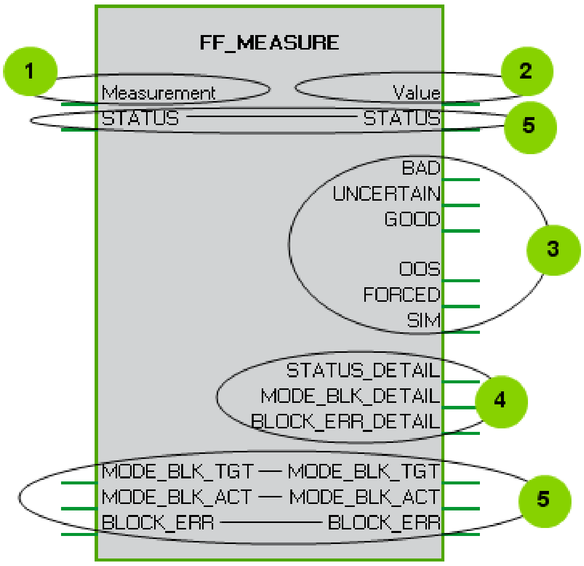


Figure 46: Measurement processing DFB

The numbered pin groups on the above figure are detailed in the following tables.

Data	Description
Measurement	Measurement value format 32 bits (IEEE-754)
StatusByte	Status bytes of the measurement value (1 byte)

Table 12: Measurement processing DFB – measurement input (1)

Data	Description
Value	Measurement value format 32 bits (IEEE-754)

Table 13: Measurement processing DFB – measurement value (2)

Data	Description
BAD	The value is not valid
UNCERTAIN	The quality of the value is less than normal, but the value may still be useful
GOOD	The quality of the value is good Possible alarm conditions may be indicated by the sub-status Check the detailed status structure to know if the value is *Good Cascade* or *Good Non-Cascade*
OOS	The function block is out of service (not running)
FORCED	The function block is in manual mode – the measurement value is set by the operator
SIM	Simulation enabled for this block

Table 14: Measurement processing DFB – process status (3)

Data	Description
BAD.Active	The status BAD is active, the value is not valid
BAD.SubStatus.NS	Non-specific
BAD.SubStatus.CE	Configuration error detected
BAD.SubStatus.NC	Not connected
BAD.SubStatus.DF	Device failure detected
BAD.SubStatus.NoComm_SF	Sensor failure detected
BAD.SubStatus.NoComm_LUV	No communication with last usable value
BAD.SubStatus.NoComm_NUV	No communication with no usable value
BAD.SubStatus.OOS	Out of service
UNCERTAIN.Active	The status *UNCERTAIN* is active, the value could be out of range or inconsistent
UNCERTAIN.SubStatus.NS	Non-specific
UNCERTAIN.SubStatus.LUV	Last usable value
UNCERTAIN.SubStatus.SV	Substitute value
UNCERTAIN.SubStatus.IV	Initial value

Data	Description
UNCERTAIN.SubStatus.SNCA	Sensor conversion not accurate
UNCERTAIN.SubStatus.EURV	Engineering unit rate violation
UNCERTAIN.SubStatus.S	Sub-normal
GOOD_NonCascade.Active	The status *GOOD NON-CASCADE* is active
GOOD_NonCascade.SubStatus.NS	Non-specific
GOOD_NonCascade.SubStatus.ACBA	Active control block alarm
GOOD_NonCascade.SubStatus.AAA	Active advisory alarm
GOOD_NonCascade.SubStatus.ACA	Active critical alarm
GOOD_NonCascade.SubStatus.UBA	Unacknowledged block alarm
GOOD_NonCascade.SubStatus.UABA	Unacknowledged advisory block alarm
GOOD_NonCascade.SubStatus.UCBA	Unacknowledged critical block alarm
GOOD_Cascade.Active	The status *GOOD CASCADE* is active
GOOD_Cascade.SubStatus.NS	Non-specific
GOOD_Cascade.SubStatus.IA	Initialization acknowledge
GOOD_Cascade.SubStatus.IR	Initialization request
GOOD_Cascade.SubStatus.NI	Not invited
GOOD_Cascade.SubStatus.NSel	Not selected
GOOD_Cascade.SubStatus.DNSel	Do not select
GOOD_Cascade.SubStatus.LO	Local override
GOOD_Cascade.SubStatus.FSA	Fault state active
GOOD_Cascade.SubStatus.IFS	Initiate fault state detected
LIMITS.OK	The measurement value is inside the limits
LIMITS.LOW_LIMITED	The value has reached its low limits
LIMITS.HIGH_LIMITED	The value has reached its high limits
LIMITS.CONST	Value forced by the operator (*MODE_BLK* in manual)
TARGET.OOS	The *Out Of Service* mode is set
TARGET.IMan	The *Initialization Manual* mode is set

Data	Description
TARGET.LO	The *Local Override* mode is set
TARGET.Man	The *Manual* mode is set
TARGET.Auto	The Automatic mode is set
TARGET.Cas	The *Cascade* mode is set
TARGET.RCas	The *Remote Cascade* mode is set
TARGET.ROut	The *Remote Output* mode is set
ACTIVE.OOS	The *Out Of Service* mode is active
ACTIVE.IMan	The *Initialization Manual* mode is active
ACTIVE.LO	The *Local Override* mode is active
ACTIVE.Man	The *Manual* mode is active
ACTIVE.Auto	The *Automatic* mode is active
ACTIVE.Cas	The *Cascade* mode is active
ACTIVE.RCas	The *Remote Cascade* mode is active
ACTIVE.ROut	The *Remote Output* mode is active

Table 15: Measurement processing DFB –detailed information (4)

Data	Description
STATUS	Status byte of the measurement value (1 byte)
MODE_BLK_TARGET	MODE_BLK Target of the measurement function block (1 byte)
MODE_BLK_ACTIVE	MODE_BLK Active of the measurement function block (1 byte)
BLOCK_ERR	BLOCK ERROR of the measurement function block (1 UInt)

Table 16: Measurement processing DFB – function block status (5)

FG-110 statistics

The *FF_FG110_STATISTICS* DFB was specifically developed to provide easy access to the Modbus statistics registers of the FG-110 linking device. These registers also contain information about the role of a redundant gateway (primary or secondary).

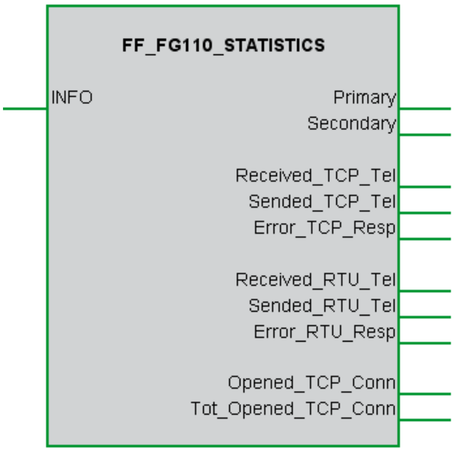


Figure 47: FG-110 statistics DFB

The following tables describe the inputs and outputs of this DFB:

Data	Description
INFO	Array of 18 bytes retrieved by the I/O Scanner of the NOC module, containing the available statistics information of the FG-110
Primary	bit = 1 if the FG-110 acts as a primary gateway in a FG-110 redundant architecture
Secondary	bit = 1 if the FG-110 acts as a secondary gateway in a FG-110 redundant architecture
Received_TCP_Tel	Number of received TCP Modbus telegrams
Sended_TCP_Tel	Number of sent TCP Modbus telegrams
Error_TCP_Resp	Number of TCP error responses
Received_RTU_Tel	Number of received RTU Modbus telegrams
Sended_RTU_Tel	Number of sent RTU Modbus telegrams
Error_RTU_Resp	Number of RTU error responses
Opened_TCP_Conn	Number of current open TCP connections
Tot_Opened_TCP_Conn	Number of total opened TCP connections

Table 17: -110 statistics DFB – I/Os description

AALARM

The objective of the *AALARM* DFB (part of the DPL) is to evaluate timed alarms associated with an analog signal. The DFB provides alarm functions by level (very high, high, low and/or very low) and by deviation in relation to a set-point value. This evaluation can be activated or deactivated individually according to the specific needs of the process:

- From the continuous and sequential control strategies implemented in the controller
- From the monitoring system

The main functions of the DFB are summarized below:

Function	Description
Level alarm	Evaluates timed alarms by level. The alarm activation can be timed. Timing and/or hysteresis can also be applied to the alarm deactivation.
Deviation alarm	Evaluates the maximum deviation alarm for the measurement from a setpoint.
Enable	Allows alarm monitoring to be enabled or disabled individually.

Table 18: Alarm DFB functions

AINPUT

The objective of the *AINPUT* DFB (part of the DPL) is to condition an analog signal normally coming from a physical input. The main functions of the DFB are summarized below:

Function	Description
Scaling	Scales the input signal (normally in raw data) to engineering units by means of a linear function.
Cut-off	Configures the value below which the measurement of the transmitter generating the input signal is not considered as reliable. If the measurement value falls below this value, the minimum measurement value is used instead.
Diagnostics	Manages the diagnostics status of the signal, if the peripherals used provide this signal, and assigns the value that is configured for failure scenarios (normally associated with the worst case scenario).
Simulation	The DFB can be configured to enter the value that should be used (in engineering units) manually. This option is usually used during tests performed from the monitoring subsystem.
External PV	The DFB allows the connection of a signal that is already in engineering units (and therefore does not need scaling, diagnosis or cut-off operations), maintaining the simulation function.

Table 19: Analog input DFB functions

DINPUT

The objective of the *DINPUT* DFB (part of the DPL) is to condition a digital signal normally coming from a physical input. The DFB provides timing functions for the connection and/or disconnection, simulation and for considering the signal state as an alarm trigger.

The main functions of the DFB are summarized below:

Function	Description
Timing	The input signal is timed so that quick state changes of the input signal are absorbed (a debounce function).
Alarm	The DFB makes it possible to enable/disable the alarm evaluation and the status that should be considered as an alarm, as well as to incorporate an external logic that should be added to the evaluation of this alarm.
Diagnostics	Manages the diagnostics status of the signal (if the used peripherals provide this signal) and assigns the value that is configured for failure scenarios (normally associated with the worst case scenario).
Simulation	The DFB can be configured to enter the value that should be used (in engineering units) manually. This option is usually used during tests performed from the monitoring subsystem.

Table 20: Digital input DFB functions

MAINPUT1

The objective of the *MAINPUT1* DFB (part of the DPL) is to condition up to four analog signals which normally come from physical inputs, as well as to select one of them based on the chosen selection criterion. The range can be configured from the monitoring subsystem.

The main functions of the DFB are summarized below:

Function	Description
Scaling	Scales the input signals to engineering units by means of a linear function. The range of the signal in engineering units can be externally configured (normally from the monitoring subsystem) within the maximum range configured from the program in the controller.
Cut-off	The block allows configuration of the value below which the measurement of the transmitter generating the input signal is not considered to be reliable – if the value falls below this signal value, the minimum measurement value is used instead. The cut-off value, in engineering units, can be externally configured (normally from the monitoring subsystem) within the maximum range that is configured from the program in the controller.
Diagnostics	The block manages the diagnostics status of the signal (if the used peripherals provide this signal) and assigns the value that is configured in case of a failure detection (normally associated with the worst case scenario). The failure value, in engineering units, can be externally configured (normally from the monitoring subsystem) within the maximum range that is configured from the program in the controller.
Simulation	The block can be configured to enter the value that should be used (in engineering units). This option allows tests (from the monitoring subsystem) on the programming associated with the block.
Selection	The block selects the analog signal from the input signals according to one of the following criteria: • Priority • Average • Direct selection • Minimum • Median • Maximum

Table 21: Multiple analog input functions

CONDSUM1

The *CONDSUM1* DFB (part of the DPL), which has features similar to those of the *CONDSUM* DFB, evaluates up to seven conditions in order to implement interlocking strategies, i.e. the logical *OR* of these conditions. Each condition can require individual resetting or be ignored (bypassed) according to the configuration of the block.

The main functions of the DFB are summarized below:

Function	Description
Interlock evaluation	The block computes a logical *OR* between all the conditions, giving condition 1 priority over condition 2, condition 2 priority over condition 3 and so on.
Safe position evaluation	Each condition has a discrete safe position associated with it so that the block evaluates which position is the safe position on the basis of the conditions that are active, and, once they have been evaluated, according to priority.
Interlocks featuring manual reset	Process conditions configured this way require resetting from the monitoring system.
Bypass	Makes it possible to bypass interlock conditions one by one – the block allows you to configure which conditions can be bypassed and which cannot (safety interlocks).
Active interlock indication	The block features output signals that report the state of each interlock condition once all the internal logic has been applied (resetting and bypassing) – these signals can be used to implement program logic or to select the analog safe position for other blocks that require it by using an external multiplexer.

Table 22: Conditions sum DFB functions

CVALVE

The objective of the *CVALVE* block is to manage valve controllers with optional position feedback (position and/or limit switches).

The main functions of the DFB are summarized below:

Function	Description
Owner	The block manages which control system level (operator or program) is the owner – as a result, it is responsible for setting the setpoint and activation for the valve controller.
Interlocking	The block makes it possible to assign the defined safe position when an active interlock is detected – an interlock bypass function is available.
Setpoint	The block makes it possible to work under a remote (normally set from the continuous control) or local (set from the program or by the operator, depending on the active owner) setpoint.
Simulation	The target position of the valve is used as the current position in simulation mode – the position limits, which are normally determined on the basis of the limit switches, are simulated on the basis of the analog position.
Tracking	The block makes it possible to activate monitoring of the actual position (in relation to the target position).

Table 23: Valve management DFB functions

4.4.2. SCADA

When building the SCADA application, Vijeo Citect objects must be defined, along with the associated data and data type.

The following recommendations are provided to facilitate design, readability and re-use:

- Exchanges are done via DDT variables when defined on the DFB
- The data dictionary links the variables in the PAC to OFS (OPC Factory Server)
- OFS allows use of Unity Pro structured variables in an unlocated format in the SCADA system

For each Genie, the corresponding DDT is used with the proper DFB. In the examples below, several Genies are presented, some of which are part of Schneider Electric's DPL while others have been specifically developed for this project.

Device and Process Library Genies	Specifically developed Genies
aiipva_10 arrow_10 mainput_1 hc_valve2_10	FG110 Status Block

Table 24: Genies list

FG-110 statistics

The FG-110 statistics Genie shows the statistic information available in the FG-110 device. The following figure and table show the Genie:

Figure 48: FG-110 statistics Genie

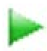
Icon	Description
 and Connexion Error	These blinking icons are shown when communication with the FG-110 is disturbed (no valid data).
	This icon is shown when communication is active and undisturbed.
	When clicking on this icon, a new window opens with detailed information about the statistics.

Table 25: FG-110 statistics Genie elements

The figure on the right shows the corresponding SuperGenie, which is displayed by clicking on the image of the FG-110 of the Genie.

The two first LEDs show the status of the device in a FG-110 redundant architecture. The third LED shows the communication status with the device. The other indicators strictly contain statistics information about Modbus communications of the device.

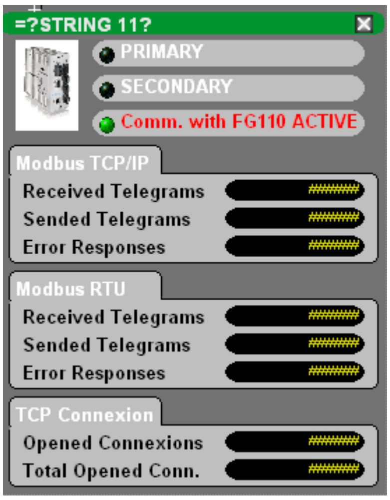


Figure 49: FG-110 statistics SuperGenie

FOUNDATION Fieldbus status

The FOUNDATION Fieldbus status Genie shows the STATUS information linked to a FOUNDATION Fieldbus measurement. The following figure and table show the graphical environment of the Genie.

Figure 50: FG-110 status Genie

Icon	Description
ST	This icon is shown when the STATUS byte linked to a measurement is *GOOD_NonCascade* or *GOOD_Cascade*.
ST	This blinking icon is shown when the STATUS byte linked to a measurement is *BAD*.
ST	This blinking icon is shown when the STATUS byte linked to a measurement is *UNCERTAIN*.

Table 26: FG-110 status Genie elements

The figure and table below show the corresponding SuperGenie, which is displayed by clicking on the Genie:

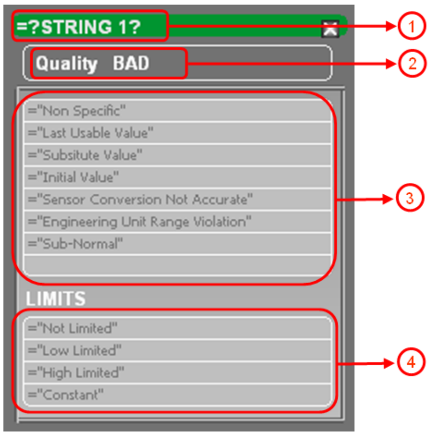


Figure 51: FG-110 status SuperGenie

Ref.	Description
1	Shows the name of the STATUS linked to the declared tag.
2	Shows the Measurement Quality Status.
3	Depending on the Measurement Quality Status, this section shows the proper possible sub-status values.
4	This section shows the Measurement Limit Conditions.

Table 27: FG-110 status SuperGenie elements

Note: The content of this SuperGenie is provided according to the Status Attribute Definition of the FOUNDATION Fieldbus Specifications FF-890-1.

FOUNDATION Fieldbus function block information

The function block which delivers the measurement is usually controlled by a specific FOUNDATION Fieldbus function block. The FOUNDATION Fieldbus function block information Genie shows the status of the *MODE_BLK* parameter (target and active) as well as the *BLOCK_ERR* status. The following figure and table show the graphical environment of the genie:

BLK

Figure 52: FOUNDATION Fieldbus function block information Genie

Icon	Description
BLK	This icon is shown when the value of the *BLOCK_ERR* tag is zero. Upon clicking this icon, a new window opens with detailed information about the *MODE_BLK* and *BLOCK_ERR*.
BLK (yellow)	This icon flashes when the value of the *BLOCK_ERR* tag is 4096, corresponding to the simulation flag. When clicking on this icon, a new window opens with detailed information about the *MODE_BLK* and *BLOCK_ERR*.
BLK (red)	This icon flashes when the value of the *BLOCK_ERR* tag is not 0 or 4096. When clicking on this icon, a new window opens with detailed information about the *MODE_BLK* and *BLOCK_ERR*.

Table 28: FOUNDATION Fieldbus function block information Genie elements

The following figure shows the graphical environment of the SuperGenie, which is displayed by clicking on the any Genie:

1 – *Mode* tab for information related to the *MODE_BLK* tag

2 – Information about the *MODE_BLK* (target and actual)

3 – *Err* tab for information related to the *BLOCK_ERR* tag

4 – Information about the errors detected on the block

Figure 53: FOUNDATION Fieldbus function block information SuperGenie

DPL Genies

The following Genies can be instantiated from the DPL:

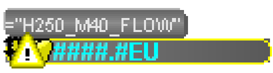
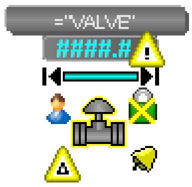
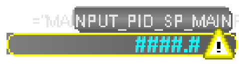
aiipva_10	arrow_10	hc_valve2_10	mainput_1
			

Table 29: DPL Genies

For further information about these genies, please refer to the DPL User Manual for Vijeo Citect.

5. Configuration

This chapter describes the configuration software's essential characteristics, main functions and data exchange methods. The following topics are described:

- FG-110 detailed configuration
- PAC detailed configuration
- Asset management software overview

The following figure shows the different steps to configure the system – the two top left items are detailed in this chapter:

Figure 54: Configuration steps

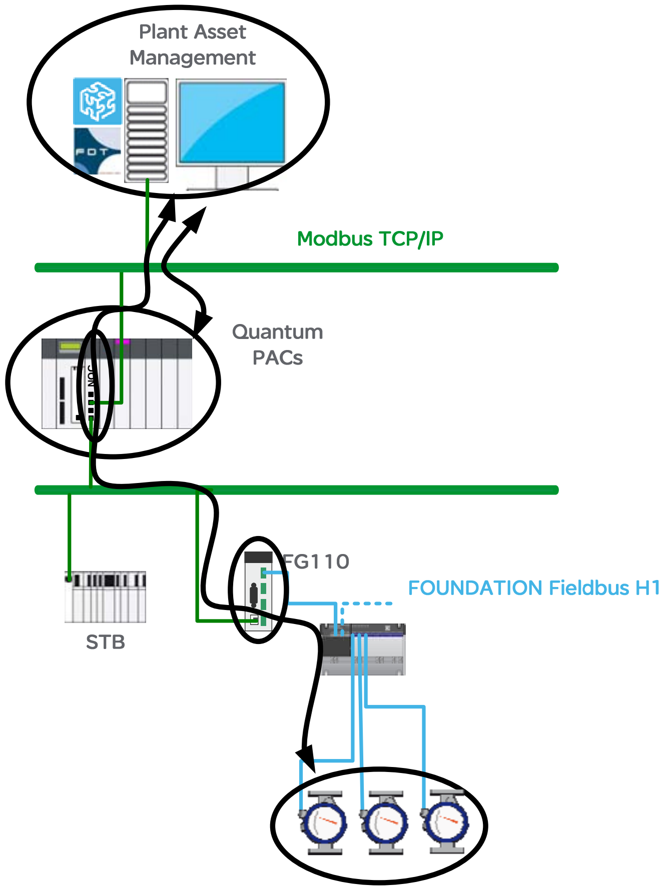


Figure 55: Instruments and FG-110

The instruments are connected to the FG-110 device FOUNDATION Fieldbus H1 port. The FG-110 is connected to the device network through its Ethernet port. The PAC is connected to the device network using port 4 of the NOC module. It is also connected to the control network using port 2 of the NOC module, which allows the asset management station to connect to the instrumentation.

Configuration of an instrument connected to the H1 port of the FFG-110 requires five steps using Unity Pro:

- Declaration and configuration of the FG-110 in the DTM browser
- Declaration and configuration of the instrument as a FG-110 slave in the DTM master
- Declaration and configuration of the NOC in the DTM browser (this operation must be performed only once)
- Declaration of a generic Modbus TCP slave on the NOC
- Addition of the FG-110 generic Modbus slave in the NOC I/O Scanner

Note: Configuration of the FG-110 (Ethernet and FOUNDATION Fieldbus H1 parameters) and the associated slaves can be performed entirely from the DTM master built into Unity Pro and the FG-110 embedded web server.

5.1. FOUNDATION Fieldbus master configuration

Configuration of the FG-110 can be divided into three main steps:

1. Configuration of the IP address and Modbus TCP/IP port behavior using a web browser.
2. FOUNDATION Fieldbus network configuration and function block application using the FF-CONF tool developed by Softing. FF-CONF allows the import of the EDD files to configure the FOUNDATION Fieldbus network. The FOUNDATION Fieldbus H1 device addresses are also assigned using FF-CONF –the address cannot be set using the FG-110 DTM.
3. Modbus mapping using a web browser.

These steps are described in the following subsections.

5.1.1. FG-110 gateway configuration parameters

The following table shows the minimum steps required to configure the FG-110 IP address and the Modbus TCP/IP port features:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Access the FG-110 linking device in a web browser using its default IP address and enter the login and password – please refer to the user manual for the default values if you have not changed them yet</td></tr>
<tr><td>2</td><td>

Access the IP configuration by clicking on *Configuration*

Click on *Internet Protocol*

Change the IP address and netmask (172.20.2.175 / 255.255.0.0)

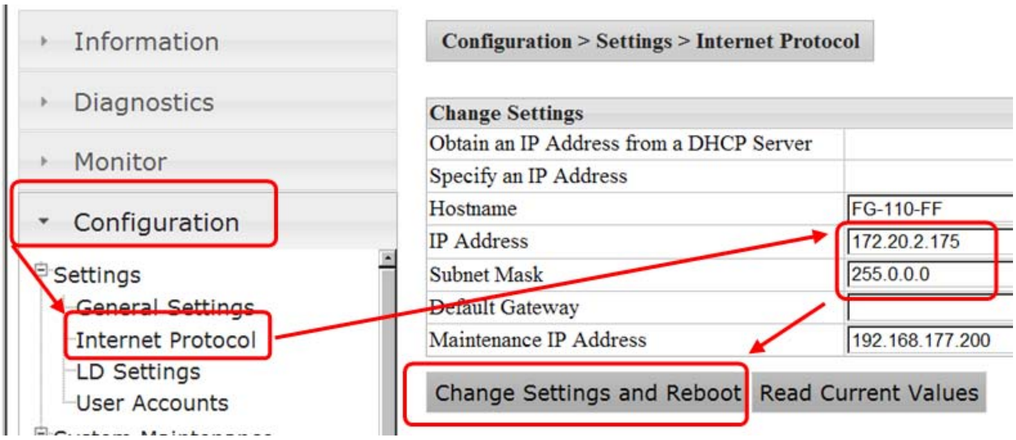


Figure 56: FG-110 configuration – IP address and mask

Click on the button *Change Settings and Reboot* to apply the new IP configuration

The FG-110 performs a software reboot

</td></tr>
<tr><td>3</td><td>Access the FG-110 linking device using its new IP address and login to the device</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>4</td><td>

Access the Modbus port configuration by clicking on *Configuration*, *Modbus* and finally *Communication*

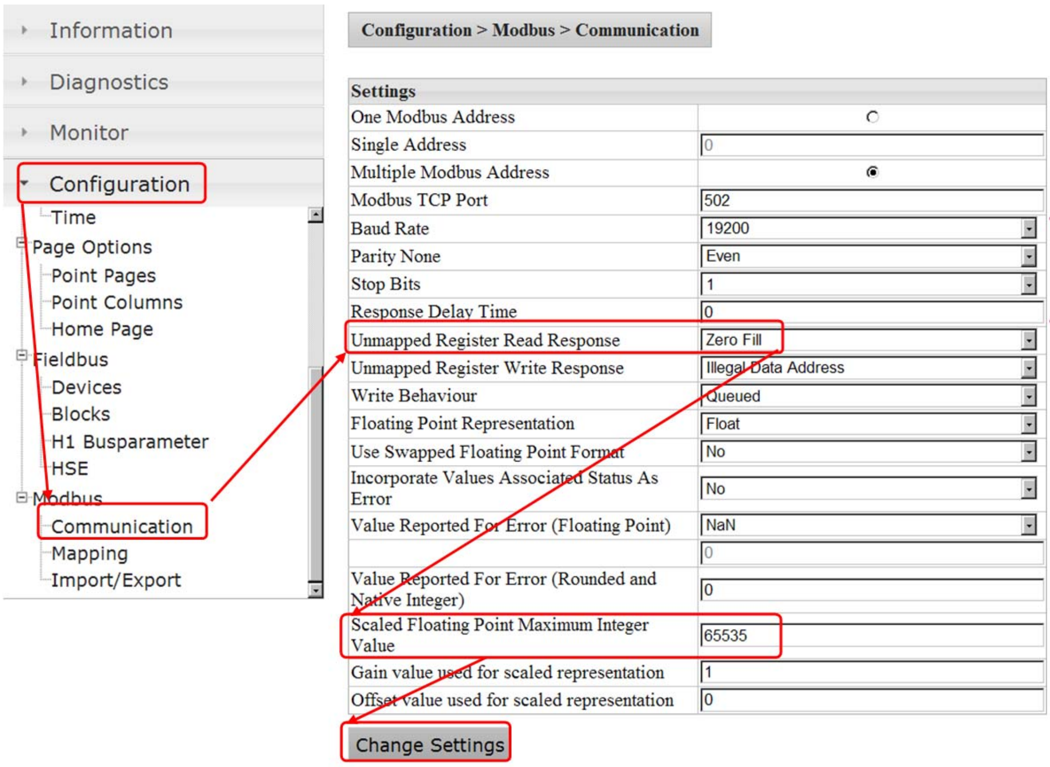


Figure 57: FG-110 configuration – Modbus parameters

Configure the following parameters:

- Unmapped Register Read Response: Zero Fill
- Scaled Floating Point Maximum Integer Value: 65535

</td></tr>
<tr><td>5</td><td>Apply the settings by clicking on the Change Settings button at the bottom of the window</td></tr>
<tr><td>6</td><td>Log out of the web session and close the web browser</td></tr>
</table>

The parameters related to the Modbus port configuration should be reviewed and adapted to the intended application, specifically those dedicated to the *Unmapped Register Response* settings (read and write).

5.1.2. FG-110 FOUNDATION Fieldbus LAS configuration

The following tables shows the main configuration steps of the FG-110 for the project detailed in this document.

Note: Although the main steps are shown, some minor actions or functionalities are skipped. For detailed information about the full configuration steps and features of the gateway, please refer to the gateway manual.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Open FF-Conf and create a new project

Two icons appear in the Network Configuration tab, representing the following items:
• The computer where the software is installed
• The IP corresponding port

Both icons show the IP address used by the computer

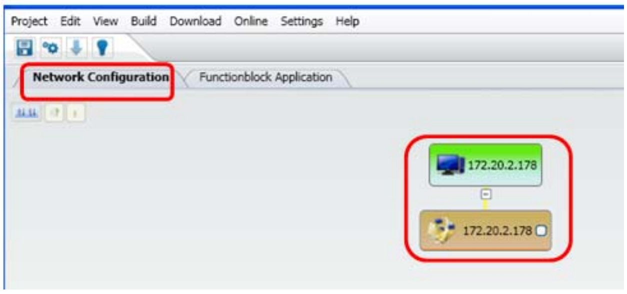


Figure 58: FF-Conf – new project</td></tr>
<tr><td>2</td><td>Add the FG-110 linking device to the project using the tool bar on the right

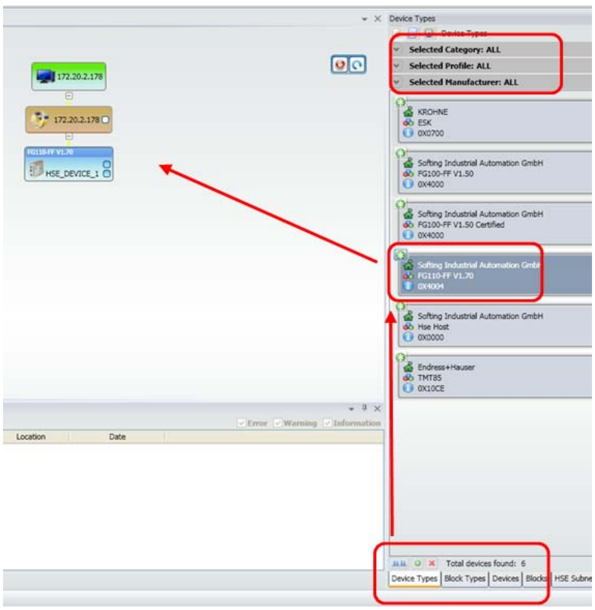


Figure 59: FF-Conf – add FG-110</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>3</td><td>
Setup the gateway IP address and change the *UserTag*, if desired

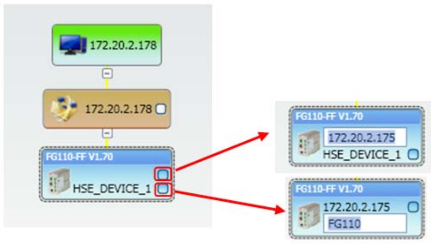


Figure 60: FF-Conf – configure FG-110 IP address
</td></tr>
<tr><td>4</td><td>
Add the H1 segment to the linking device

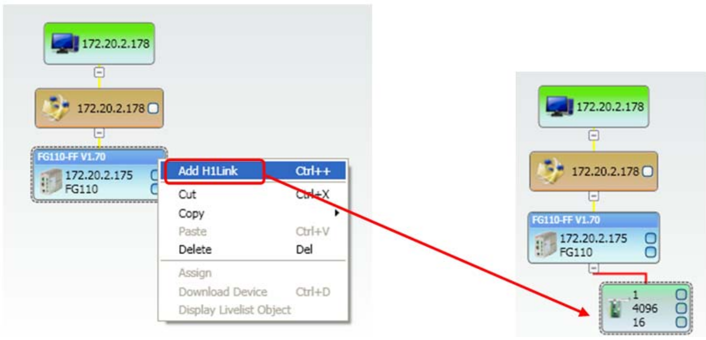


Figure 61: FF-Conf – add HA segment

In our example, we will use the H1 port number 1 on the FG-110 linking device
</td></tr>
<tr><td>5</td><td>
Install the DD files of the instruments (refer to the manual of the gateway for the detailed procedure) and include them into the project

Select the HS1 port and click on the green icon of each device (tool bar on the right) to assign this kind of device to the port

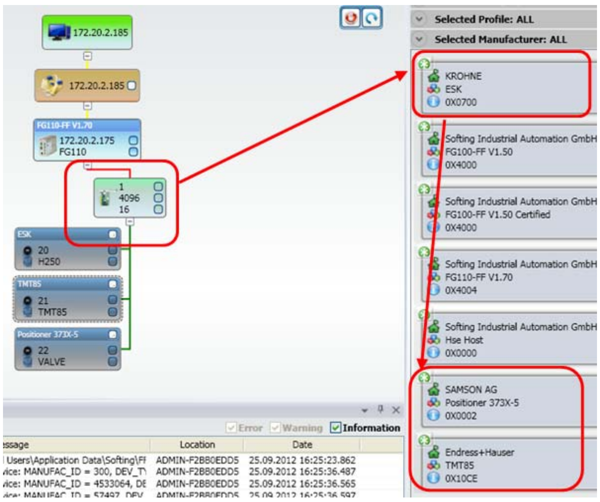


Figure 62: FF-Conf – add instruments
</td></tr>
</table>

Schneider Electric

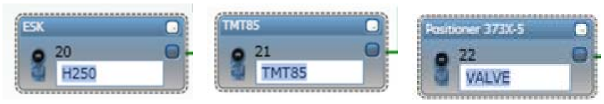
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>6</td><td>Set the NodeID (bus address) and the UserTag for each device


Figure 63: FF-Conf – configure devices</td></tr>
<tr><td>7</td><td>The physical configuration (i.e. FG-110 port usage and device addresses) is now finished

Assign this configuration to the existing devices on the network – this process requires the following steps:
• Open the Network Livelist window
• Once all the available devices are displayed, select the FG-110 linking device
• Open the Network Configuration window, right-click the FG-110 linking device to open a contextual menu and select Assign
• On the upper right side of the FG-110 icon, a yellow light should appear (which means that the assignment of the device is in progress) – once the assignment is successful this light changes to green
• Repeat the previous steps for each device to perform the assignment of the devices connected to the FG-110 (port and instruments of the FG-110)</td></tr>
<tr><td>8</td><td>Below is a screenshot with all devices correctly assigned


Figure 64: FF-Conf – FG-110 final assignment result</td></tr>
</table>

Once the network topology is built and the corresponding configuration is applied to the devices on the network, the next step is to build the function block application on the instruments.

A very important parameter on each function block is the *MODE_BLK* parameters. Depending on the required links for the function blocks application, the *MODE_BLK* target must be set to the proper values. If not configured properly, the blocks do not run as expected.

The following figure shows the *MODE_BLK* target value for each function block in our application example:

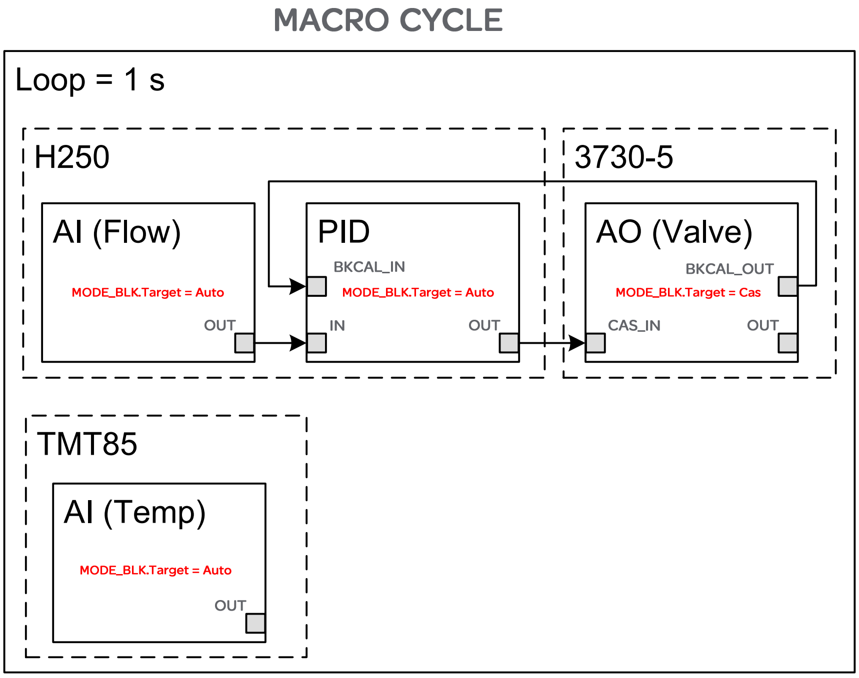


Figure 65: MODE_BLK target value

The following table describes how to build the function block application:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>

Open the *Functionblock Application* window

From the tool menu on the right, select the desired function blocks for each device – in our case the function blocks are the following:

<table>
<tr><th>Manufacturer</th><th>Instrument</th><th>Function block</th></tr>
<tr><td>KROHNE</td><td>Flow meter H250</td><td>AI
PID</td></tr>
<tr><td>E+H</td><td>Temperature sensor iTMT85</td><td>AI</td></tr>
<tr><td>SAMSON</td><td>Positioner 3730</td><td>AO</td></tr>
</table>

Table 30: Function blocks used in the example application

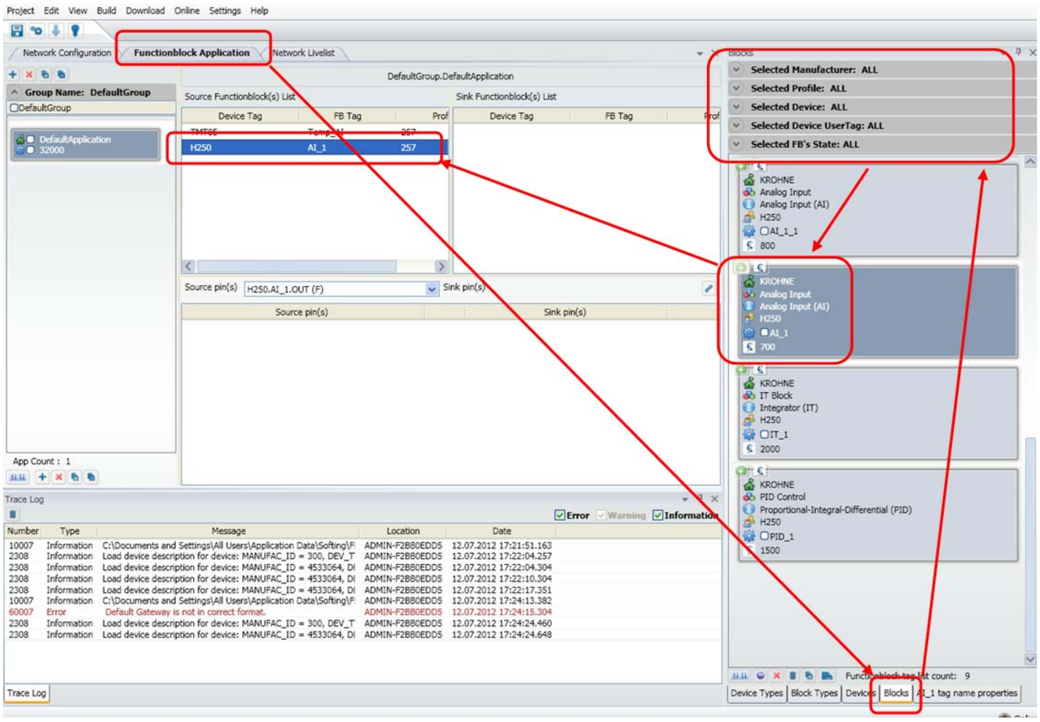


Figure 66: FF-Conf – build the function block application

Note: We only use one macro cycle with a period of one second – it is created by default and all the new function blocks are placed on this macro cycle

Note: The value for the macro cycle time is given in 1/32 of milliseconds, so the default value of one second is configured by entering the value 32000

</td></tr>
</table>

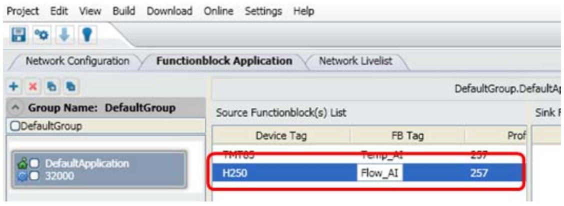
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>Edit the FB Tag, if desired, by double clicking on the function block tag name label


Figure 67: FF-Conf – edit the function block tag name</td></tr>
<tr><td>3</td><td>Once the function blocks are declared, set the links between them, if required

Select the source pin and the destination link for the function block

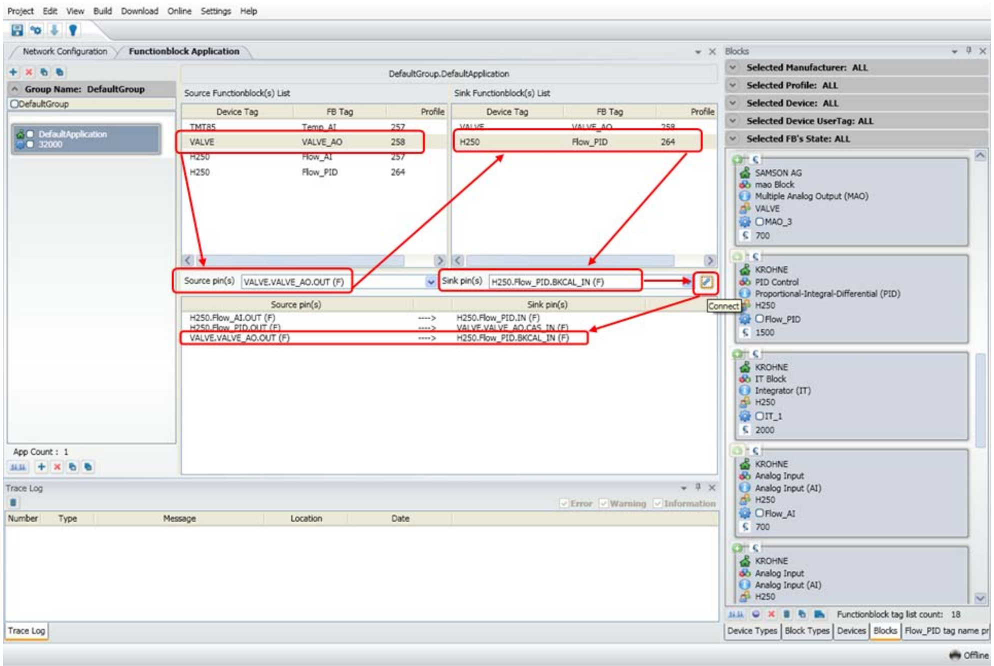
Click on the Connect button to apply this linkage


Figure 68: FF-Conf – link function blocks

The link information should appear on the links list in the center window on the screenshot</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>4</td><td>
Configure the specific parameters for each declared function block

- Right click on the function block to open the contextual menu
- Select *FB parameter view*

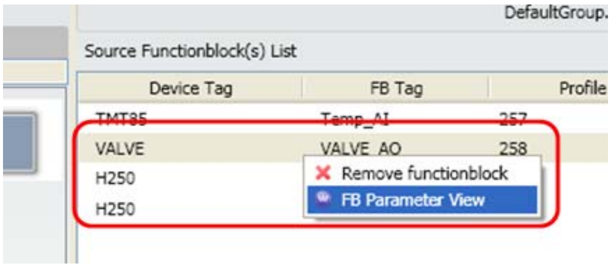


Figure 69: FF-Conf – open function block parameter view
</td></tr>
<tr><td>5</td><td>
The main parameters of the function block are related to the channel assignment (only for the measurement instruments) and the *MODE_BLK*

Set the channel and the mode block *target*, *permitted* and *normal* parameters

The following screenshot shows the configuration parameters in our project for the H250 flow meter:

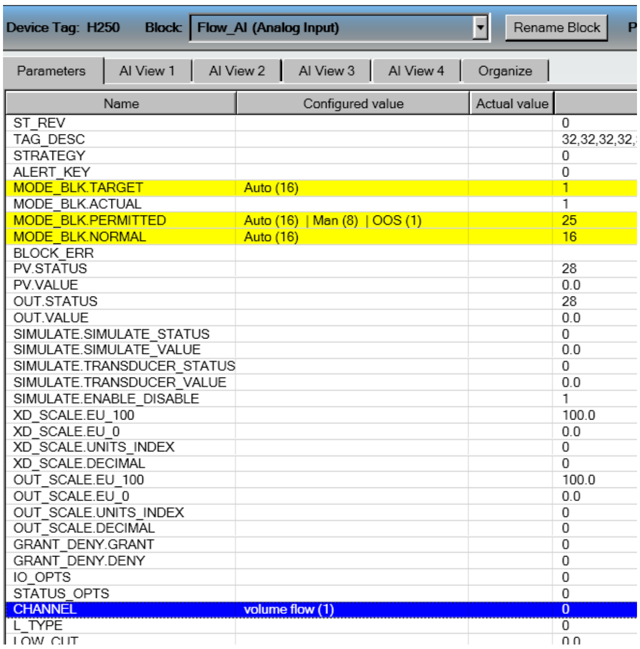
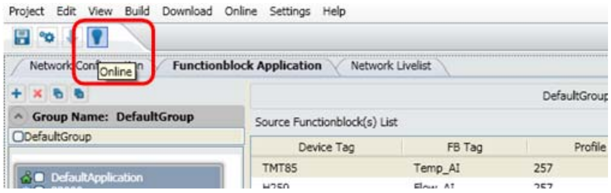
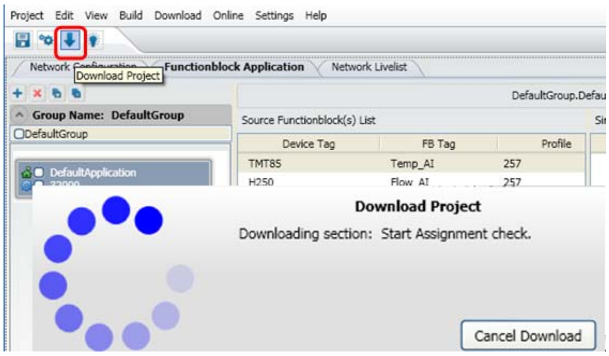


Figure 70: FF-Conf – H250 flow meter configuration

These parameters should be configured for all the function blocks declared in the project accordingly
</td></tr>
<tr><td>6</td><td>Save the project</td></tr>
</table>

Step	Action
7	If you are not in online mode, select the online mode  **Figure 71: FF-Conf – Online mode**
8	Build the project and download the function block application to the devices  **Figure 72: FF-Conf – Project building and download to the devices**

5.1.3. FG-110 Modbus mapping

The following table shows the procedure to map the information in the Modbus registers of the FG-110:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Open a web browser, connect to the FG-110 device web server and log in</td></tr>
<tr><td>2</td><td>Enter the Modbus memory map by selecting Configuration, Modbus and Mapping in the left tree view of the web browser

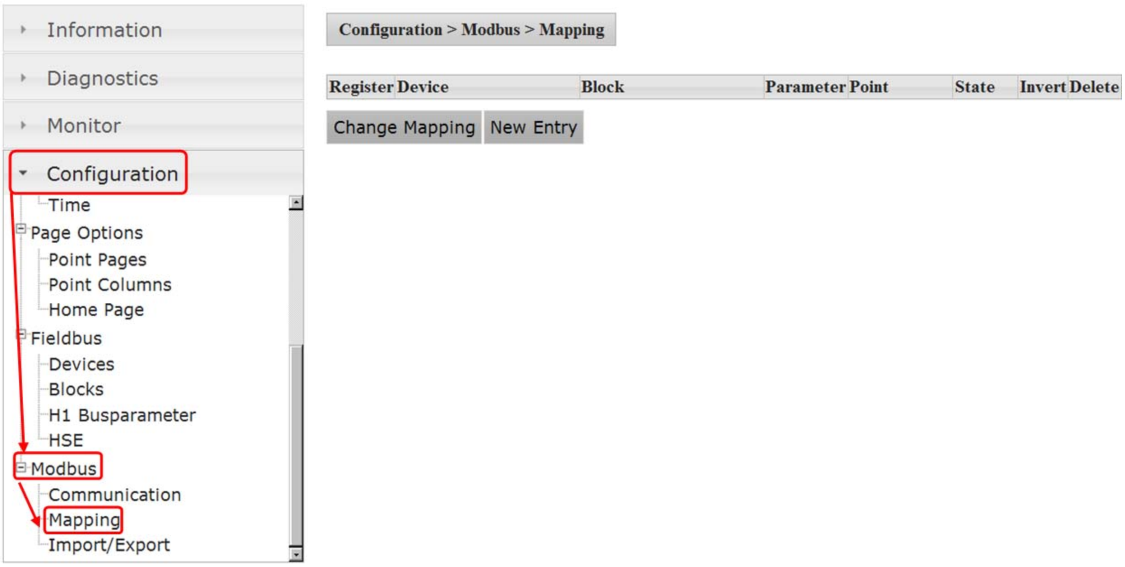


Figure 73: FF-Conf – Modbus mapping window</td></tr>
</table>

Step	Action
3	Map the following information:
4	Apply the new mapping by clicking on the *Change Mapping* button

Map the following information:

Register	Device	Block	Parameter	Point
40001	H250	Flow_AI	OUT	Flow_AI
40003	H250	Flow_AI	BlockError	Flow_AI
40004	H250	Flow_AI	OUT	.OUT.STATUS
40011	TMT85	Temp_AI	OUT	Temp_AI
40013	TMT85	Temp_AI	BlockError	Temp_AI
40014	TMT85	Temp_AI	OUT	.OUT.STATUS
40021	H250	Flow_PID	OUT	Flow_PID
40023	H250	Flow_PID	IN	Flow_PID
40025	H250	Flow_PID	BlockError	Flow_PID
40026	H250	Flow_PID	OUT	.OUT.STATUS
40031	VALVE	VALVE_AO	OUT	VALVE_AO
40033	VALVE	VALVE_AO	BlockError	VALVE_AO
40034	VALVE	VALVE_AO	OUT	.OUT.STATUS
40101	H250	Flow_AI	Mode	Flow_AI
40102	TMT85	Temp_AI	Mode	Temp_AI
40103	H250	Flow_PID	Mode	Flow_PID
40104	VALVE	VALVE_AO	Mode	VALVE_AO
40105	H250	Flow_PID	SP	Flow_PID
40107	VALVE	VALVE_AO	SP	VALVE_AO

Figure 74: FF-Conf – Modbus mapping data

Edit the *Point* right column and add the appendix *.STATUS* at the end of the text to map the Function Block status values

Remember the output measurement values are in real format, so it takes two words in the register

5.2. PAC Configuration

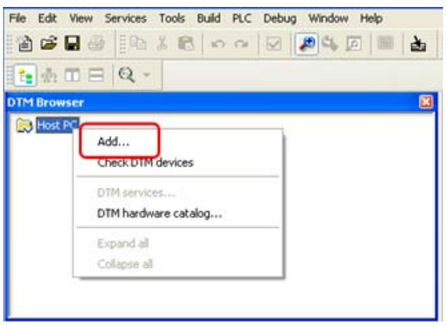
The NOC DIO Quantum module provides connectivity between the control network (SCADA applications, asset management stations, engineering stations and so on) and the device network (instruments, actuators and so on). Two main functionalities must be configured in the application and both use the DTM technology:

- FG-110 communication DTM, which allows configuration and setup of the instrumentation
- NOC Master DTM, which allows device services configuration (including I/O Scanner to multiple devices on the network, e.g. FG-110)

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION Configure the default values at each level (default measurement values if the instrument cannot send it, default command on an actuator if the communication with the PAC is lost and generally default network values) in case part of your system is not working properly. **Failure to follow these instructions can cause death, serious injury or equipment damage.**

5.2.1. Definition and configuration of the FG-110 in the Unity Pro DTM browser

The following table shows the instrument DTMs declaration using Unity Pro. Some steps (such as the installation of the DTM files on the computer) are not detailed – please refer to the proper documentation if you need more details.

Step	Action
1	Open the Unity DTM browser
2	Right click on the *Host PC* icon of the DTM browser and select *Add…*  **Figure 75: Unity Pro – Add a new DTM**

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>3</td><td>Select the FG-110 linking device FF HSE in the new window and click on Add DTM

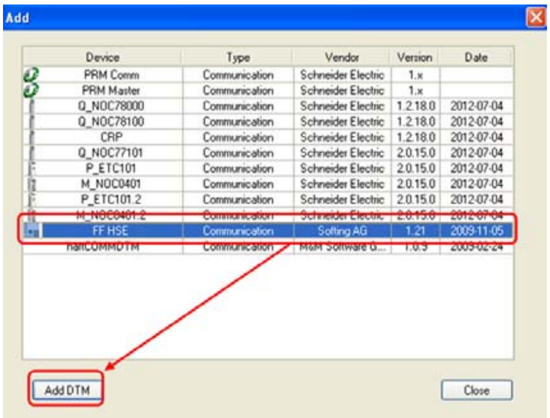


Figure 76: Unity Pro – Select the FG-110 DTM</td></tr>
<tr><td rowspan="2">4</td><td>Set the Alias Name of the device and click on OK to finish the device insertion

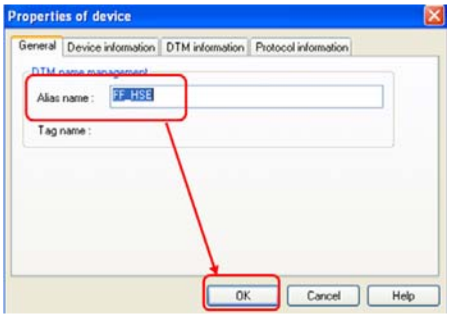


Figure 77: Unity Pro – Configure the FG-110 DTM alias name</td></tr>
<tr><td>The DTM browser displays the following icon:

Figure 78: Unity Pro – DTM browser with FG-110</td></tr>
<tr><td>5</td><td>Add the FG-110 H1 DTM by right clicking the FF-HSE DTM and selecting the Add menu

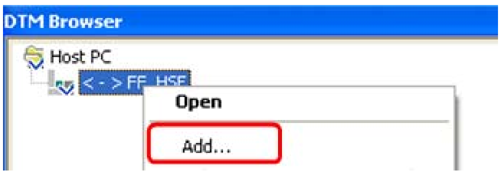


Figure 79: Unity Pro – add a new DTM</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>6</td><td>Select the FG-110 H1 DTM and click on Add DTM

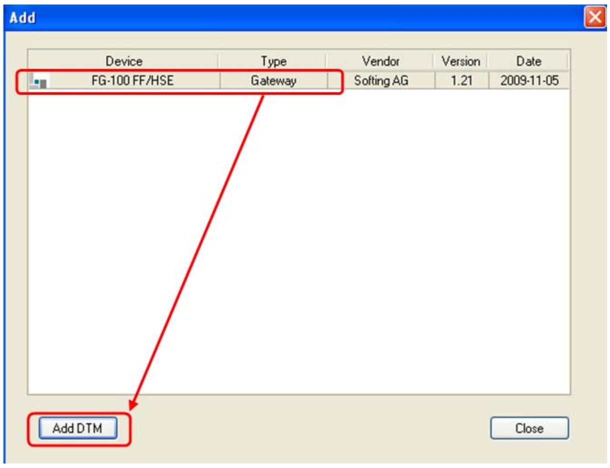


Figure 80: Unity Pro – Select the FG-110-H1 DTM</td></tr>
<tr><td>7</td><td>Set the Alias Name of the device and click on OK to finish the device insertion

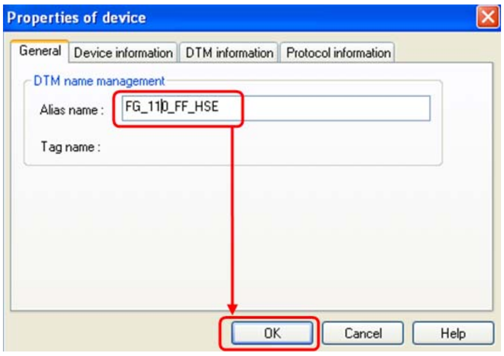
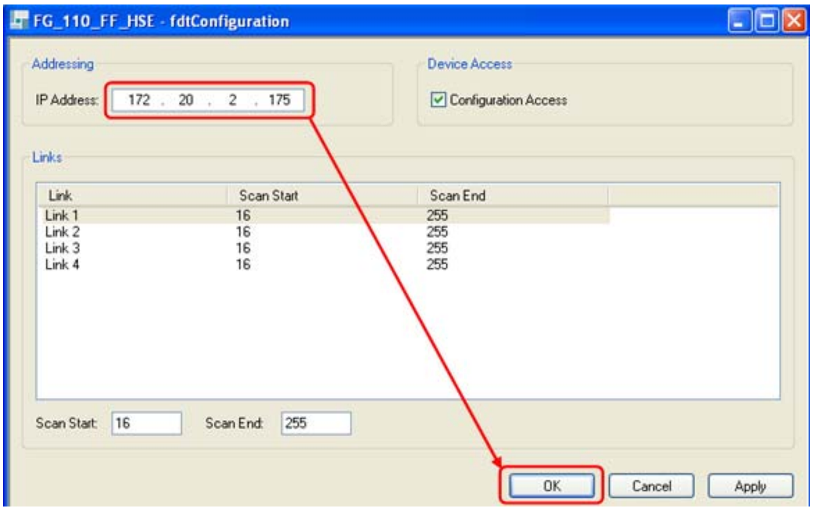


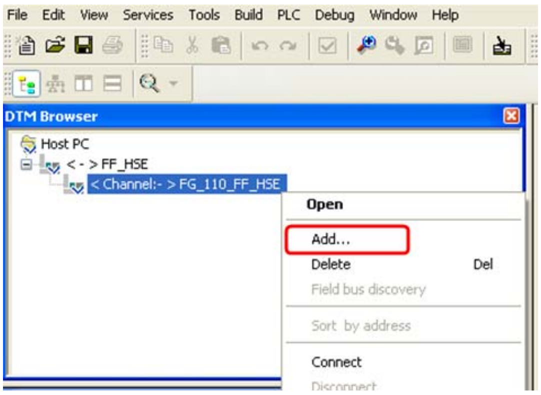
Figure 81: Unity Pro – Set the alias name for the FG-110-H1 DTM</td></tr>
<tr><td>8</td><td>The DTM browser displays the following icons:

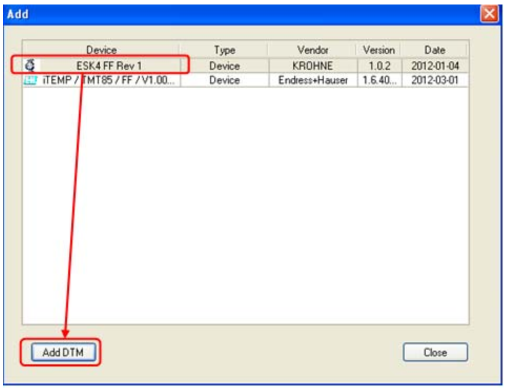
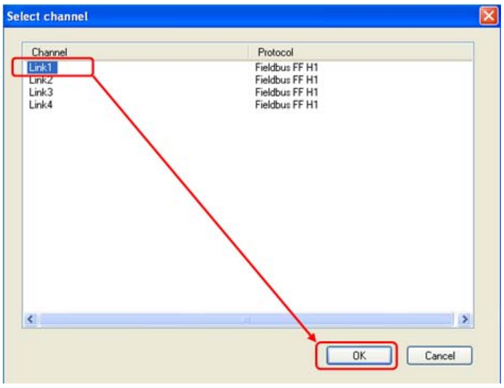
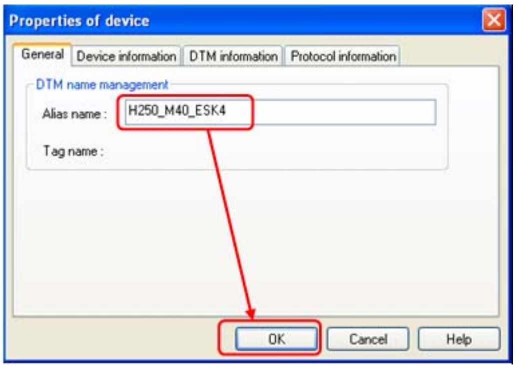
Figure 82: Unity Pro – DTM browser with FG-110-H1 DTM</td></tr>
</table>

Step	Action
9	Double click the FG-110 H1 DTM to open the device properties window Set the device IP address with the one configured on the device  **Figure 83: Unity Pro – Configure the FG-110-H1 DTM IP address** Click *OK* to finish the device DTM configuration

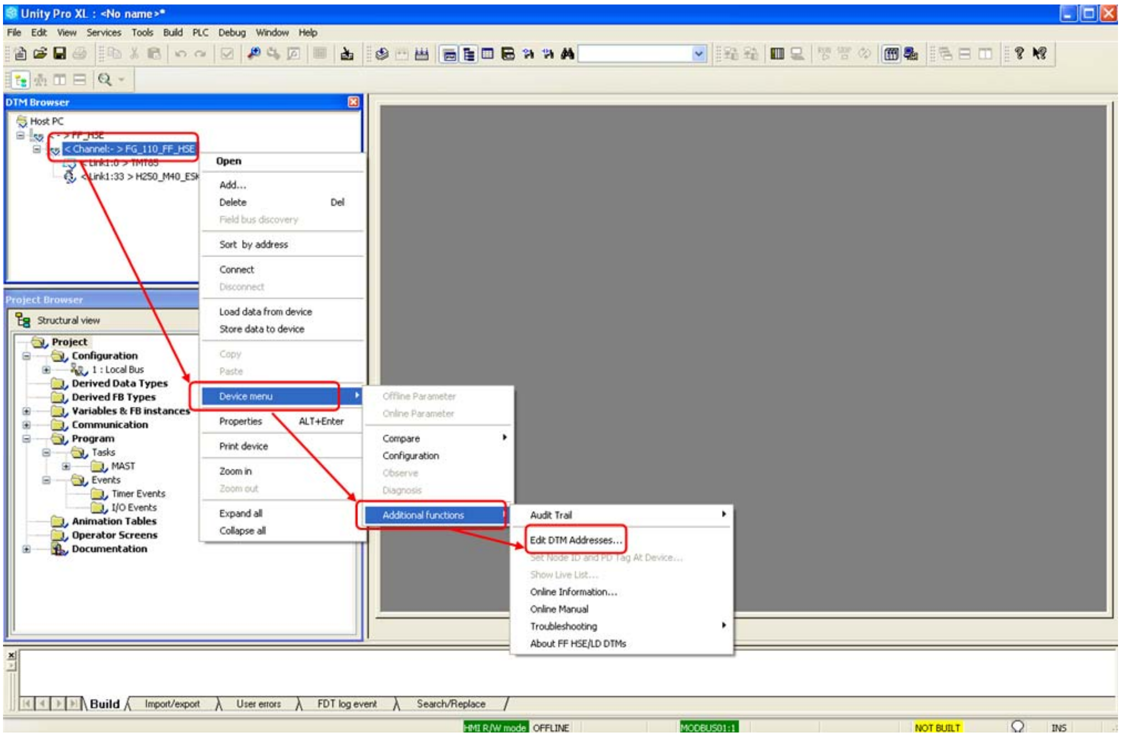
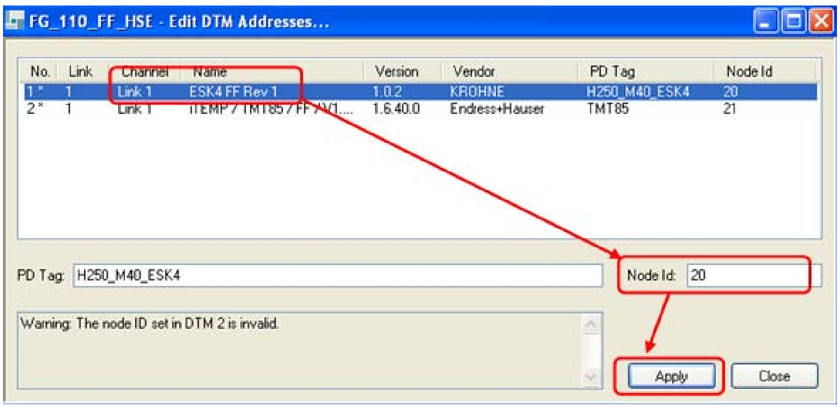
5.2.2. DTM instrumentation definition

The following table shows the procedure to declare the instrumentation DTMs in the Unity Pro DTM browser. For particular information about commissioning a specific instrument through the DTM interface, please refer to the user manual provided by the device manufacturer.

Step	Action
1	Open the Unity Pro DTM browser
2	Right click on the FG-110 channel icon and click *Add…*  **Figure 84: Unity Pro – Add a new DTM**

Step	Action
3	Select the instrument to add to the DTM configuration  **Figure 85: Unity Pro – Select the instrument DTM**
4	Select the H1 port of the gateway, which will be connected the instrument  **Figure 86: Unity Pro – Select the H1 port for the instrument**
5	Set the instrument *Alias name*, if desired, to finish the DTM declaration process  **Figure 87: Unity Pro – Configure the alias name for the instrument DTM**
6	Repeat steps 2 to 5 for each instrument in your configuration

Schneider Electric

Step	Action
7	Once all the instruments are declared in the project, configure the instrument addresses to match the addresses declared with FF-Conf. To do this, right click on the FG-110 channel icon and select *Device Menu*, *Additional Functions* and *Edit DTM Addresses…*  **Figure 88: Unity Pro – open instruments DTM addresses configuration** Remember that the addresses were assigned to the instruments previously (using FF-CONF)
8	In the new window, select the instrument and set the address  **Figure 89: Unity Pro – configure instruments DTM addresses** Once all addresses are configured, close the window by clicking *Apply*

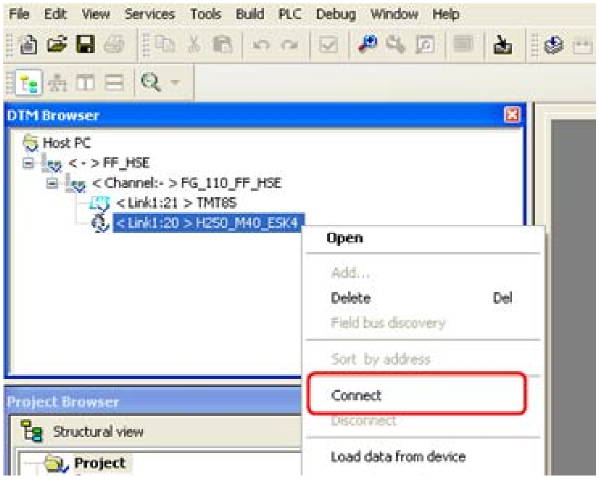
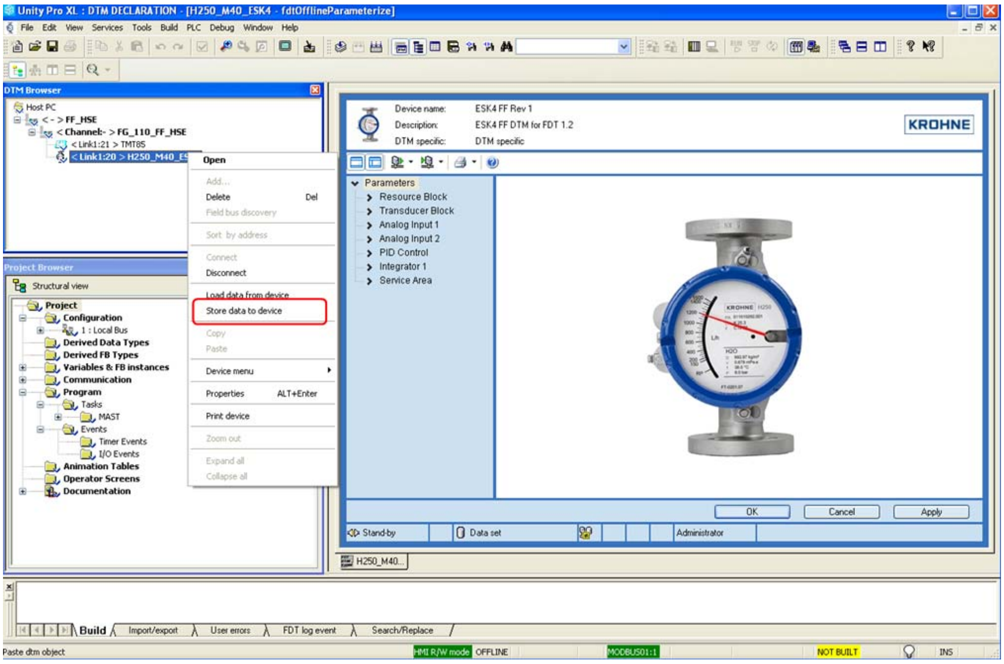
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>9</td><td>Check the communication and perform the instrument commissioning by doing the following:
• Right click on the icon of the instrument to open the contextual menu of the instrument
• Select Connect


Figure 90: Unity Pro – connect to an instrument
If the connection is successful, the icons of the FG-110 linking device, the FG-110 channel and the selected instruments appear in bold


Figure 91: Unity Pro – connected to an instrument</td></tr>
<tr><td>10</td><td>Open the instrument DTM configuration and set up all the required parameters
Note: This step is directly related to the FOUNDATION Fieldbus protocol – configure all the parameters of the function blocks according to the requirements and constraints of the FOUNDATION Fieldbus standard</td></tr>
</table>

Schneider Electric

Step	Action
11	Download the configuration to the devices by doing the following: • Connect the DTM to the instrument • Right click on the instrument icon • Select *Store data to device*  **Figure 92: Unity Pro – store data to device**
12	Repeat the steps 10 and 11 for all the instruments in your project

5.2.3. NOC configuration in the PAC

The following table describes how to configure the NOC module in the PAC using Unity Pro XL:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>
Configure the hardware by adding the NOC 78000 module to the hardware configuration

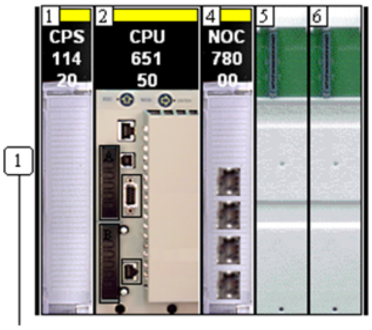


Figure 93: Unity Pro – PAC configuration

Note: the Quantum NOC78000 is available with Unity Pro V7.0 or later
</td></tr>
<tr><td>2</td><td>
Insert the NOC module in the rack configuration

The DTM is automatically included in the Unity Pro DTM browser

Set the *Alias name* of the new NOC DTM in the new window

Change the alias, if desired, and click *OK* to finish the DTM declaration

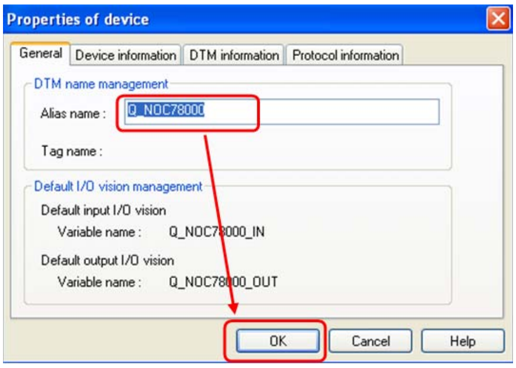


Figure 94: Unity Pro – set NOC DTM alias name

The new NOC DTM appears in the Unity DTM browser

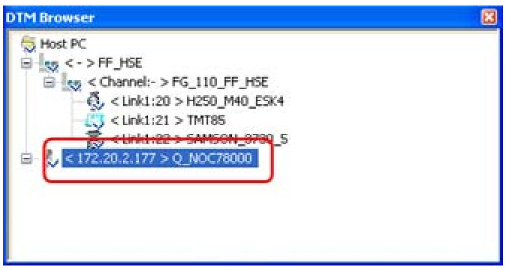


Figure 95: Unity Pro – DTM browser with NOC DTM
</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>3</td><td>
Configure the module IP address and netmask by doing the following:

Double click on the NOC icon in the DTM browser
In the opened properties window, click on Channel Properties and TCP/IP
Configure the IP address and the netmask – in our example, the IP is set to 172.20.2.177 and the netmask to 255.255.0.0
Apply the changes by clicking Apply

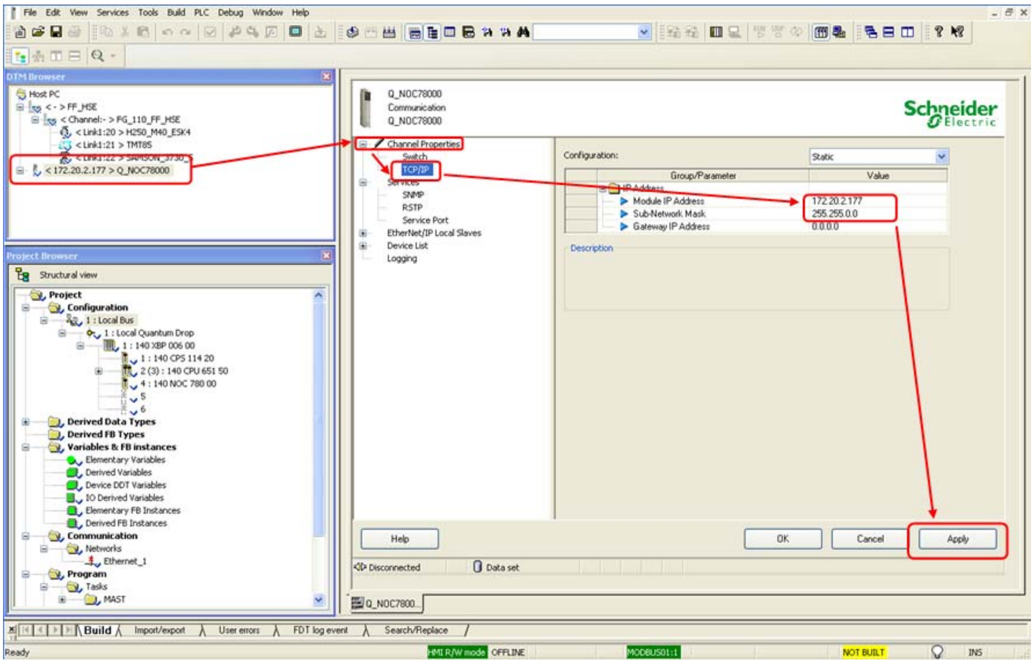


Figure 96: Unity Pro – configure NOC IP address
</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>4</td><td>

Disable the services that will not be used by the NOC. To do this:

- Select *Services* in the left tree view of the device DTM properties window
- Disable the *Address Server*, *QoS Tagging* and *Network Time Service* options
- Click *Apply* button to validate the changes

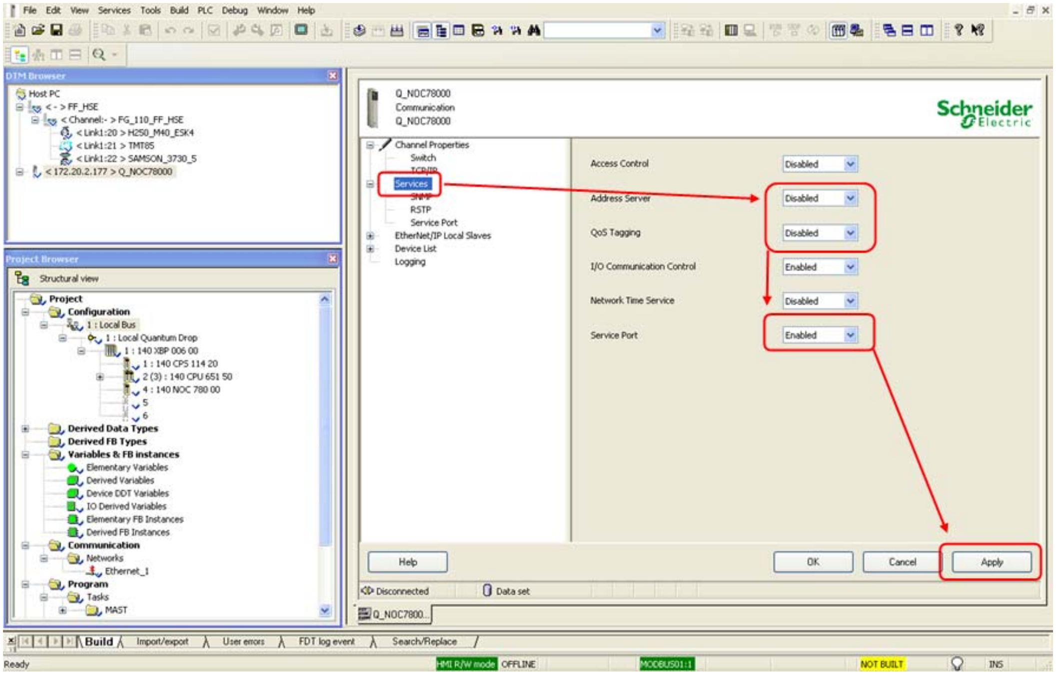


Figure 97: Unity Pro – configure NOC services

</td></tr>
<tr><td>5</td><td>

In order to use the NOC I/O Scanning service DTM, a Modbus TCP/IP device connected to the NOC module must be declared in the DTM browser – in our case, we use a generic Modbus TCP DTM pointing to the IP address of the FG-110, which allows the NOC to use the I/O Scanning service

Select the NOC icon in the Unity Pro DTM browser, right click it to open the device contextual menu and select *Add…*

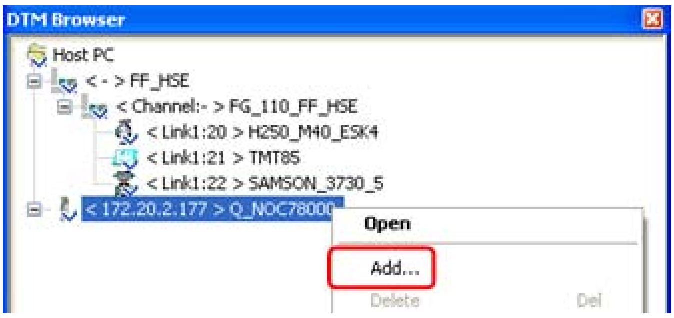
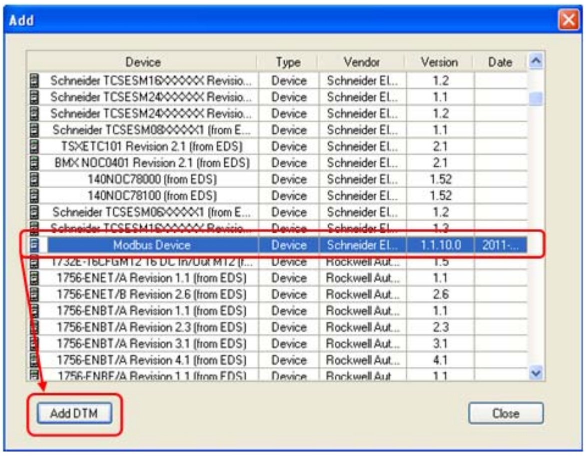
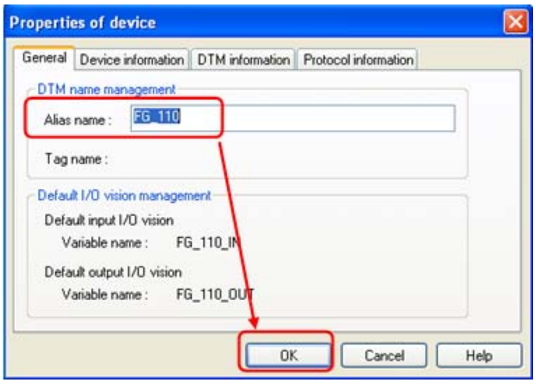


Figure 98: Unity Pro – add a new DTM

</td></tr>
</table>

Step	Action
6	Select *Modbus Device* in the list and click *Add DTM*  **Figure 99: Unity Pro – select Modbus device DTM**
7	On the device properties window, change the device *Alias name* – in our case *FG_110* **Note**: This step is recommended because this name is used later to build DDTs which contain information about the module  **Figure 100: Unity Pro – set the FG-110 DTM alias name**

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>8</td><td>
A new icon appears in the DTM browser, directly connected to the NOC

A new IP address is automatically assigned to the new module

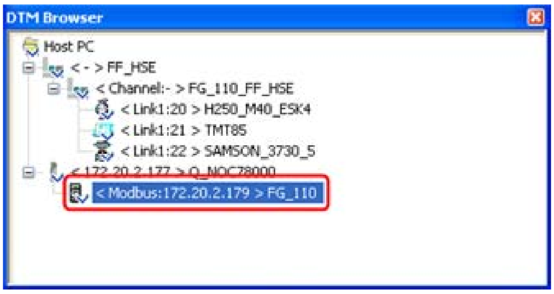


Figure 101: Unity Pro – DTM browser with FG-110

Note: The generic DTM Modbus device does not have any property to configure in its DTM
</td></tr>
</table>

Once some devices are connected to the NOC module in the DTM browser, the I/O Scanning can be configured. The first step (not mandatory) is to modify the I/O structure names created by Unity Pro.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>
Click on the *Device List* node of the DTM device properties window

Open the parameter window for the device *FG_110* (the generic Modbus device)

Click on the *Properties* tab and set up the new input/output names – in our case *FG_110_IN* and *FG_110_OUT*

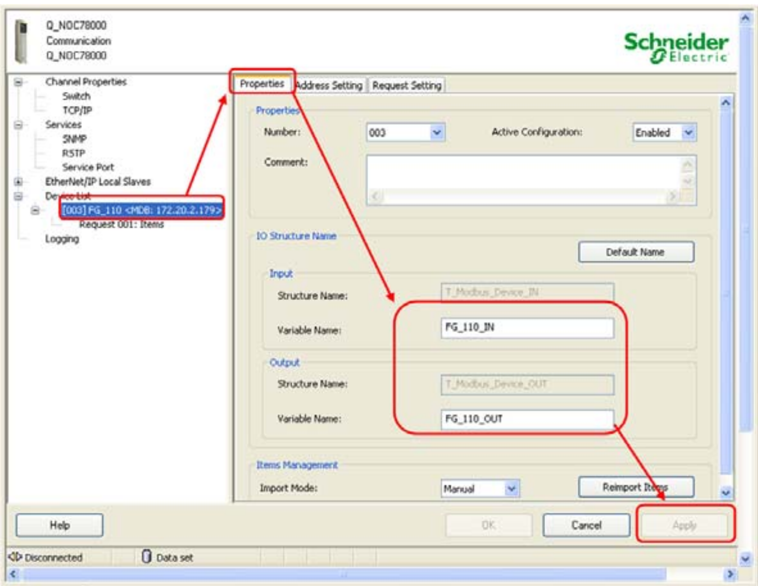


Figure 102: Unity Pro – FG-110 DTM properties

Validate the changes by clicking *Apply*
</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>

Configure the proper IP address for the generic Modbus device by activating the *Address Setting* tab

Set up the new IP address – in our case 172.20.2.175 – and validate the changes by clicking *Apply*

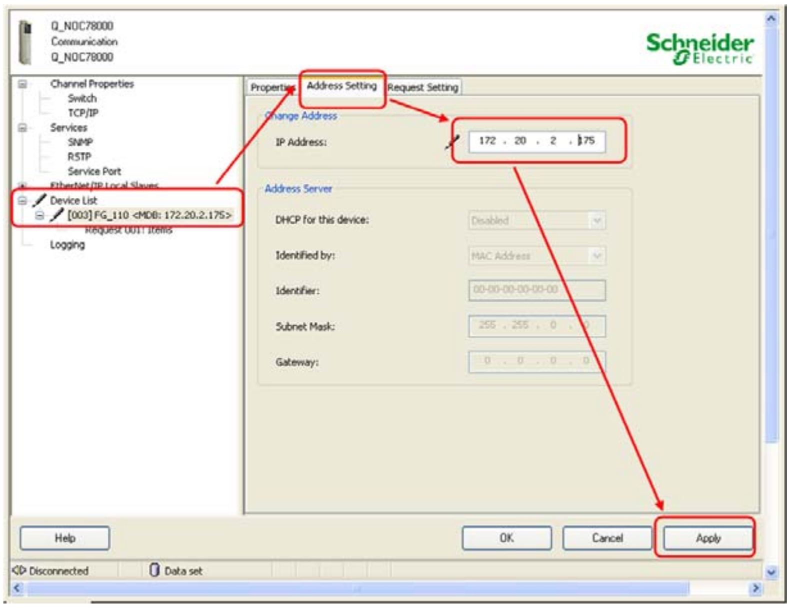


Figure 103: Unity Pro – FG-110 IP address configuration

Note: After applying changes, the IP address of the generic Modbus device in the DTM browser will change to the newly configured IP address

</td></tr>
</table>

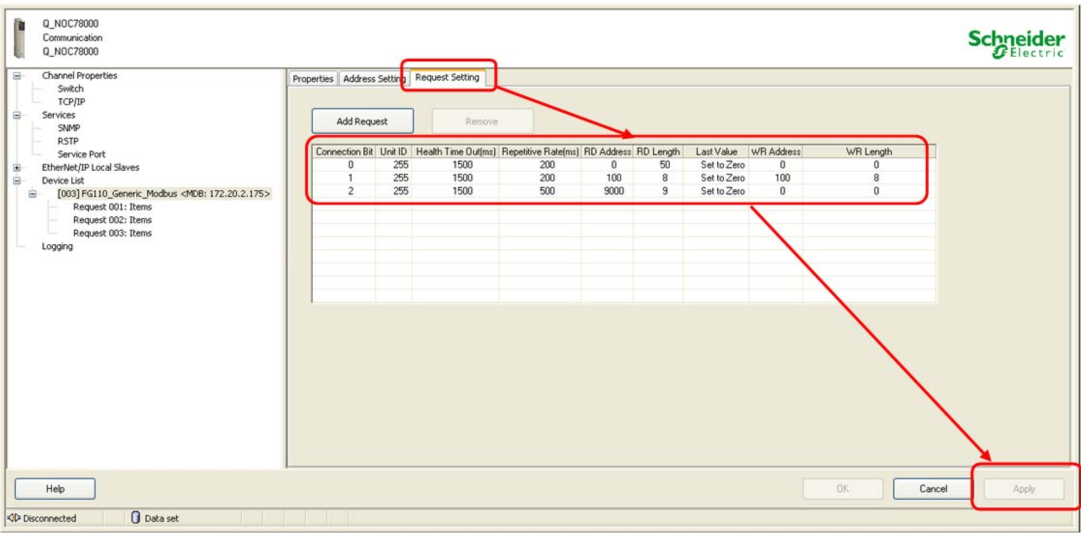
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>3</td><td>
Click on the Request Setting tab and set up the I/O Scanning line.
All the read values beginning at the address 40001 and finishing at the address 40035 have been mapped, leaving some blank zones as spare. Declare a first I/O Scanning line for this memory zone for reading – beginning at the address 40001 with a length of 50 words (first I/O Scanning line on the screenshot below)
For the read/write values, declare the data at the memory zone beginning at the address 40101 and finishing at the address 40108. Declare a first I/O Scanning line for this memory zone, merging read and write – beginning at the address 40101 with a length of eight words (second I/O Scanning line on the screenshot below)
To get the Modbus TCP/IP statistics information of the gateway (which is placed in a reserved memory zone of the gateway), declare a third I/O Scanning line to read the zone beginning at the address 49001 with length of nine words


Figure 104: Unity Pro –I/O scanning configuration
Note: Some of the mapped values have a real format, so they use two Modbus registers
</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>4</td><td>

When performing a program analysis or build operation, Unity Pro automatically creates a DDT variable which contains all the read variables in a structure. This structure can be directly adapted by creating the variables in the expected format with customized names. This operation is not mandatory, but it is highly recommended to have the project clearly organized

Once the I/O Scanning lines are defined, select the *Request properties* of the device list

Activate the *Input* tab

Highlight the first four bytes and click *Define Item(s)* to create the flow measurement variable

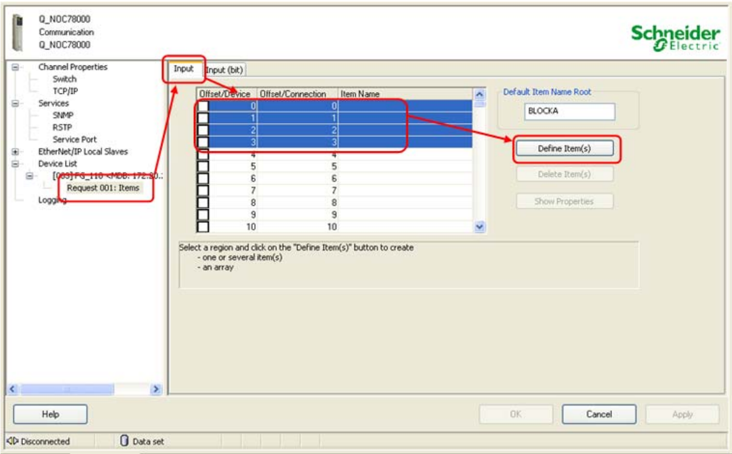


Figure 105: Unity Pro – add new item to the input table

Choose the new variable data type in the *New Item(s) Data Type* combo box (*REAL* in our case)

Set the variable name (*Flow_Measurement* in our case)

Apply the changes by clicking *OK*

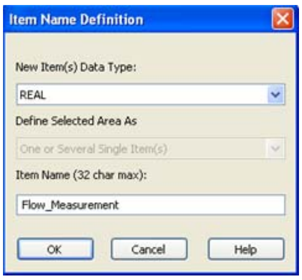


Figure 106: Unity Pro – set measurement variable name

</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>5</td><td>
Highlight the byte number 4 and click Define Item(s) to define the STATUS variable for the flow measurement

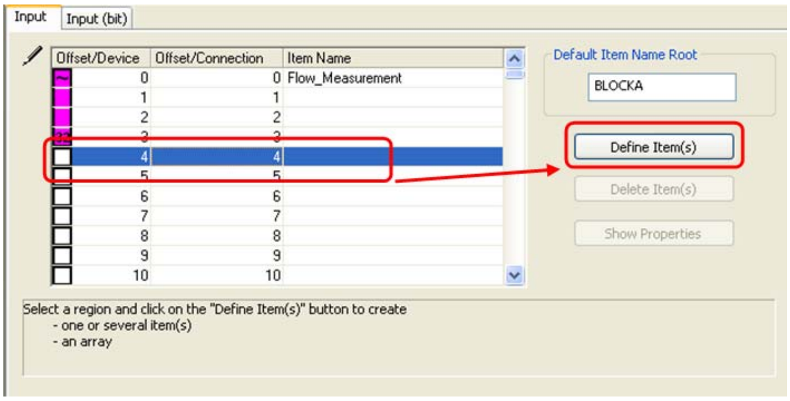


Figure 107: Unity Pro – add new item to the input table

Choose the variable data type (Byte)

Set the variable name (Flow_STATUS in our case)

Apply the changes by clicking OK

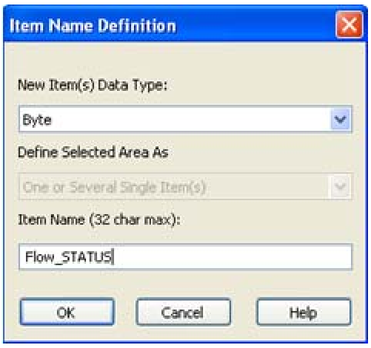


Figure 108: Unity Pro – set status variable name
</td></tr>
<tr><td>6</td><td>
The following figure shows the result of the last steps:

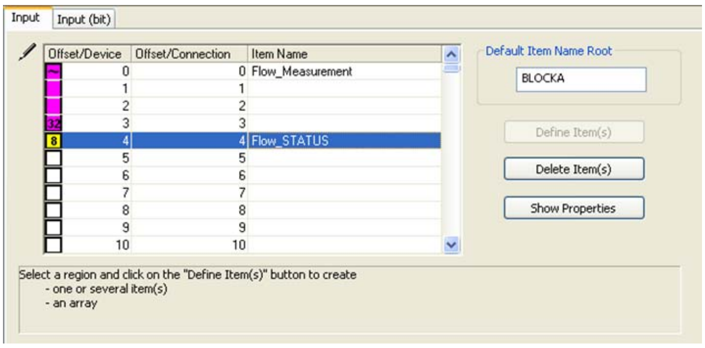


Figure 109: Unity Pro – configured flow measurement and status
</td></tr>
<tr><td>7</td><td>Repeat the steps 4 to 6 to declare all the variables in your application</td></tr>
</table>

Schneider Electric

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>8</td><td>

Define the reserved NOC memory zone on the PAC in order to exchange all the information concerning the I/O Scanning and the related control and status bits. To do this:

- Open the PAC hardware configuration
- Select the NOC module and open the NOC properties window
- Select the *Configuration* tab and set the %MW which will be used by the NOC

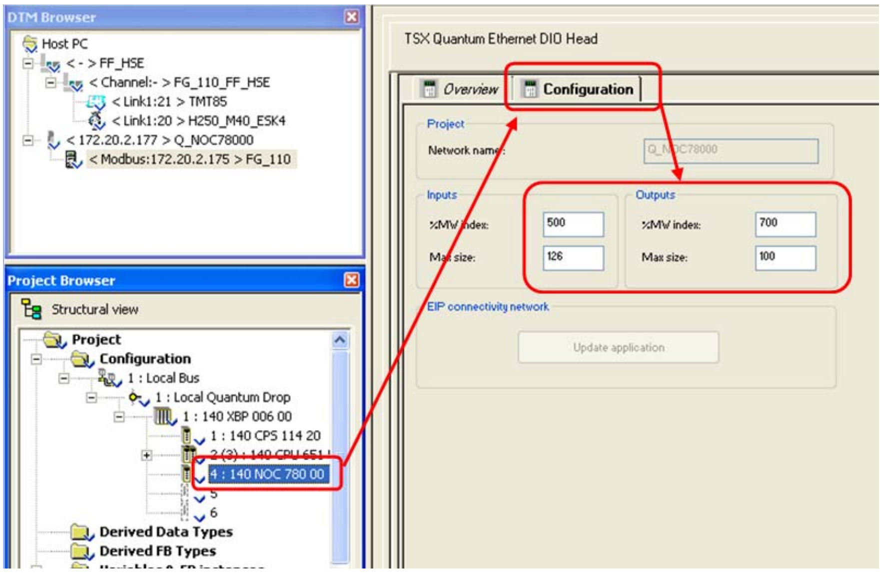


Figure 110: Unity Pro – memory zone NOC configuration

</td></tr>
<tr><td>9</td><td>

Once all the parameters are configured, perform an analysis or build operation to verify the variables which will contain the defined variable and the status and control variables for the I/O Scanning

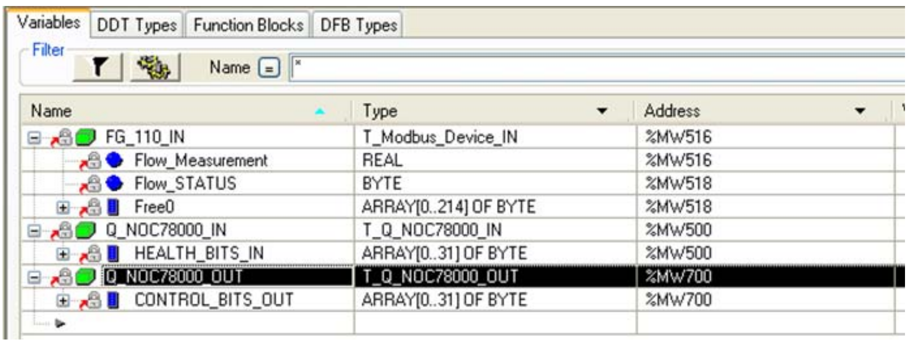


Figure 111: Unity Pro – data editor

In our example we have defined the flow measurement (*FG_110_IN.Flow_Measurement*) and the flow measurement status (*FG_110_IN.Flow_STATUS*)

The structured variable *Q_NOC78000_IN* contains the health information of each I/O Scanning line and the structured variable *Q_NOC_78000_OUT* contains the control execution information

</td></tr>
</table>

5.3. Asset management software overview

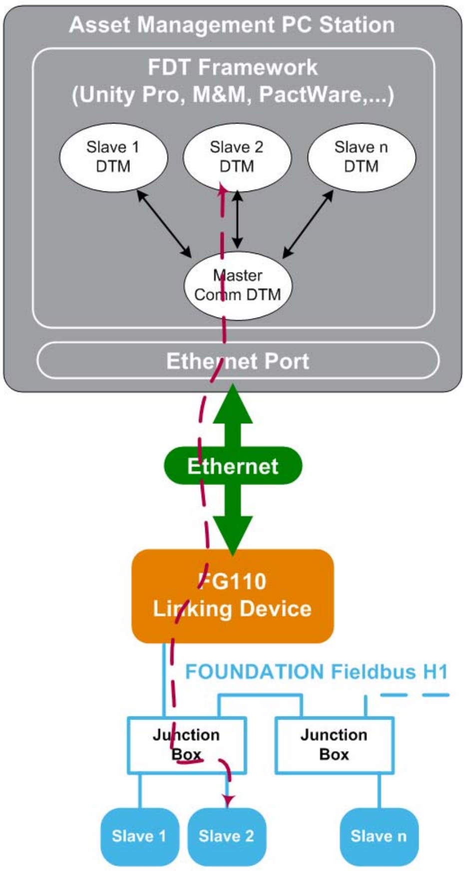


Figure 112: Asset Management connectivity

The asset management software can connect to the various devices on the bus through the FG-110 linking device. It can also define the parameters used by the equipment, such as type of unit, scaling, calibration, *MODE_BLK* parameters, alarms and reports. These settings can be made through the Ethernet port.

The software must import a DTM file to exchange data with the device (parameter setting, data reading or writing and so on). The DTM is a software driver developed by the manufacturer for a specific device. This DTM file encapsulates:

- Device-specific data
- Communication capabilities
- Graphical elements
- Simple Graphical User Interface (GUI) for functions such as configuration, operation, calibration and diagnostics
- Help files

6. Implementation

This chapter details the implementation of the components introduced in the design chapter and discusses the relationship between the implemented components and the configuration parameters.

6.1. Objectives

The aim of the project is to connect the PlantStruxure architecture to the FOUNDATION Fieldbus H1 segment through the FG-110 linking device, which also acts as a gateway between FOUNDATION Fieldbus and Modbus. The features implemented in the example application using Modbus include the following:

- Measurement processing through the AI function block
- Actuator processing through the AO function block
- PID linkage using function blocks to control an electropneumatic positioner, including the setup of the setpoint for the control by the operator through the SCADA application
- Function block diagnostics using the *BlockError* parameter and the status of the *OUT* parameter
- FG-110 linking device information, statistics and communication status

6.2. I/O Scanning failure detection

The communication between the Quantum PAC and the FG-110 is performed with the NOC Modbus TCP/IP I/O Scanning service. One of the main objectives is to monitor that this communication is active and properly executed.

The bit *Q_NOC78000_IN.HEALTH_BITS_IN[0].x* indicates that the I/O Scanning line number *x+1* of the Ethernet Port 3 is properly executed, so we use this bit to monitor the communication. In our example application, we use three I/O Scanning lines, so the used bits are *.0*, *.1* and *.2*.

In order to provide simulation capabilities in the system, we use the *DIPNUT* DFB from the DPL. Furthermore, this DFB offers time filtering functionalities. A Genie is available to represent this DFB in the SCADA.

The usage of the DFB and its parameters are represented below:

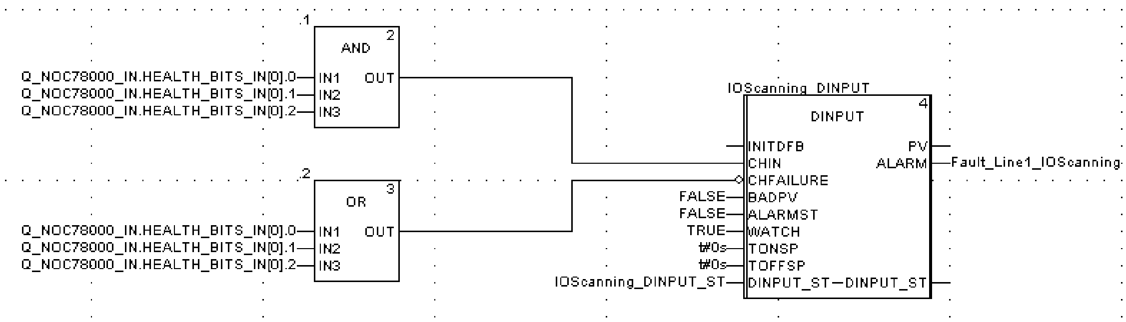


Figure 113: DINPUT DFB usage and parameters

The *Fault_Line1_IOScanning* output can be used in the instrument DFB to identify a communication issue with the FG-110 linking device.

You should activate the alarm monitoring in the SCADA so it can display the communication disturbances.

Figure 114: I/O Scanning monitoring

The Genie proposes an interface to enable the simulation mode of the healthy I/O Scanning communication signal.

6.3. Measurement processing

The combination of the *AIPNUT* and *AALARM* DFBs from the DPL provides the following functions to the AI function block:

- Scaling
- Cut-off
- Simulation
- Alarming
- Corresponding Genie for the SCADA

The *PV* output of the *AINPUT* DFB is connected to the *PV* input of the *AALARM* DFB.

The *FF_Measure* DFB gets detailed information about the FOUNDATION Fieldbus measurement and the function block delivering this measurement – it was specifically developed to decode the status bits, the *MODE_BLK* and the *BLOCK_ERR*. The input parameters of the DFB are linked to the variables created by the NOC I/O Scanning. The measurement output is linked to the external PV input (*EXTPV*) of the *AINPUT* DFB – in our application, the measurement scaling is performed by the instrument. The following data is used to provide the channel input failure (i.e. input signal no longer usable): the measurement status (values *BAD* or *UNCERTAIN*) and the *Fault_Line1_IOScanning* provided by the *DINPUT* DFB. The *_DETAIL* structures are created specifically to get diagnostics information that can be used on the application – All the information is decoded bit by bit and then reacts depending on the issue and the process needs.

The usage of these DFBs and their links is described below:

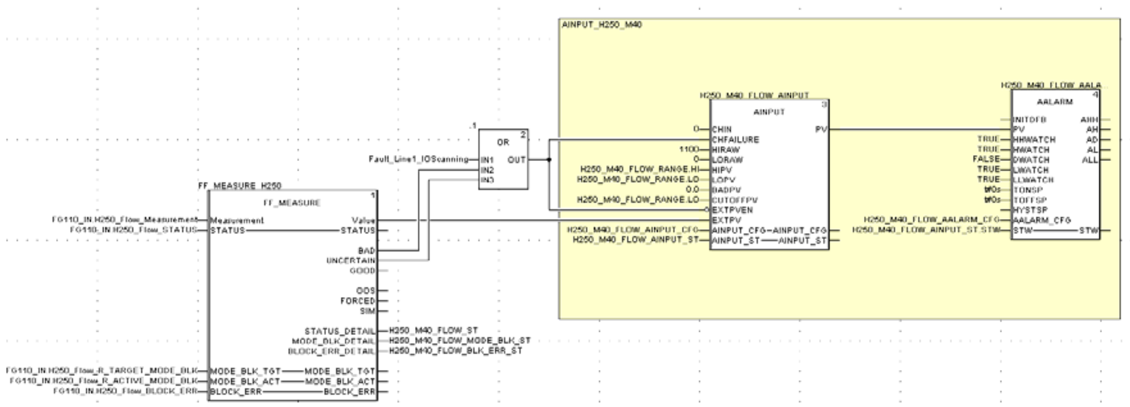


Figure 115: Measurement processing implementation

Note: In a real process application, the PV output of the *AINPUT* DFB should be used by your process.

The corresponding Genie displays the measurement value (treated by the *AINPUT* DFB) in the SCADA application. You should configure the threshold and the alarm monitoring (not like those of the instrument defined in the FOUNDATION Fieldbus standard) so the Genie can display the alarm state. The genie also has an interface to simulate a value in the PAC (not linked to the simulation function of the instrument).

The following figure shows Genie interface for the H250 flow meter AI measurement:

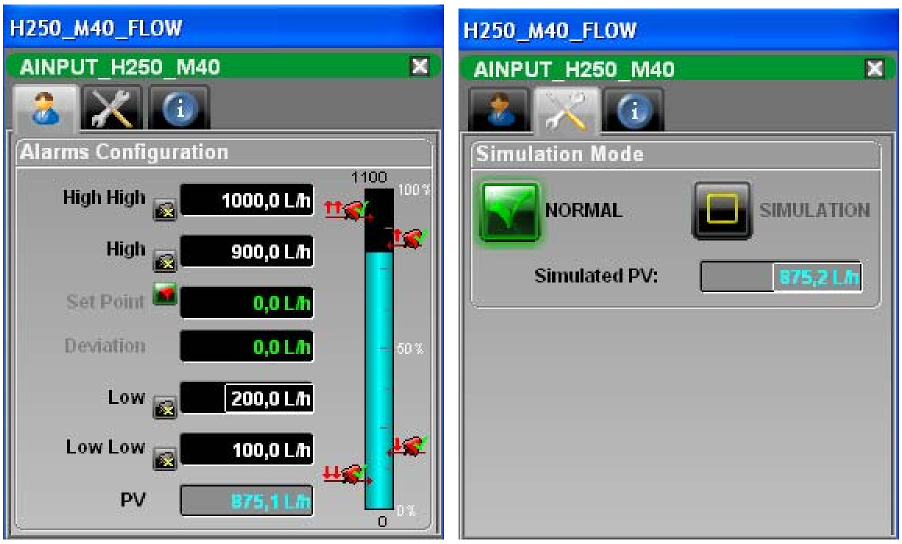


Figure 116: Measurement processing SuperGenies

6.4. Actuator processing

A PID function block embedded on the H250 flow meter is used to control the valve positioner. The *MAINPUT1* DFB of the DPL provides several options to change the setpoint of the PID.

The *PV* output of this DFB is directly assigned to the variable mapped on the I/O Scanning DDT to write the PID setpoint.

The usage of this DFB is depicted below:

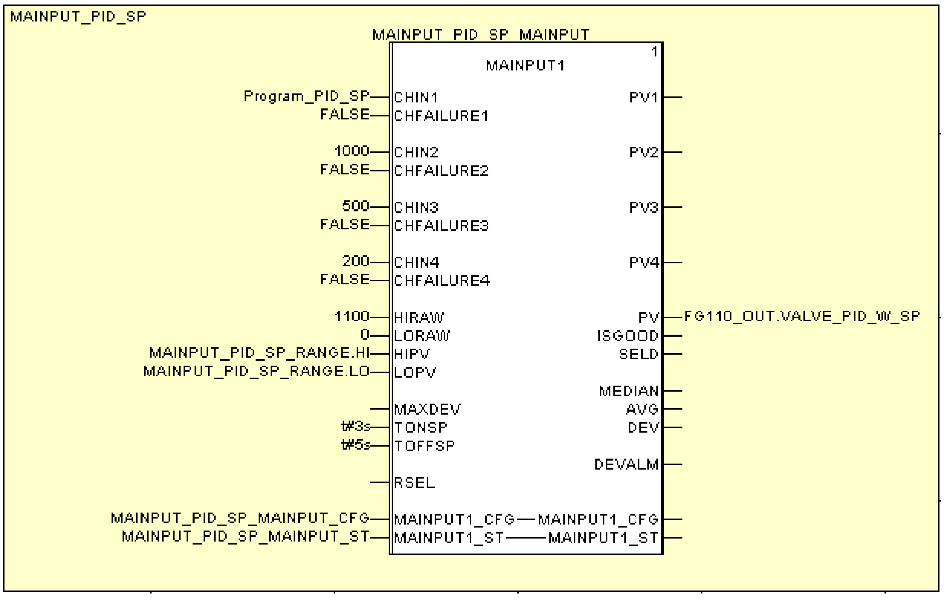


Figure 117: MAINPUT DFB used with the FOUNDATION Fieldbus PID

Using the corresponding Genie on the SCADA, the system operator can change the PID setpoint manually, select one of the predefined values on the PAC, use a value calculated by the PAC or select a mix of them.

The corresponding SuperGenie is shown below:

Figure 118: MAINPUT SuperGenie used with the FOUNDATION Fieldbus PID

We use the *FF_Measure* DFB to check the PID output value. This DFB decodes the information related to the OUT status and the PID MODE_BLK parameter for diagnostics purposes.

The usage of this DFB is depicted below.

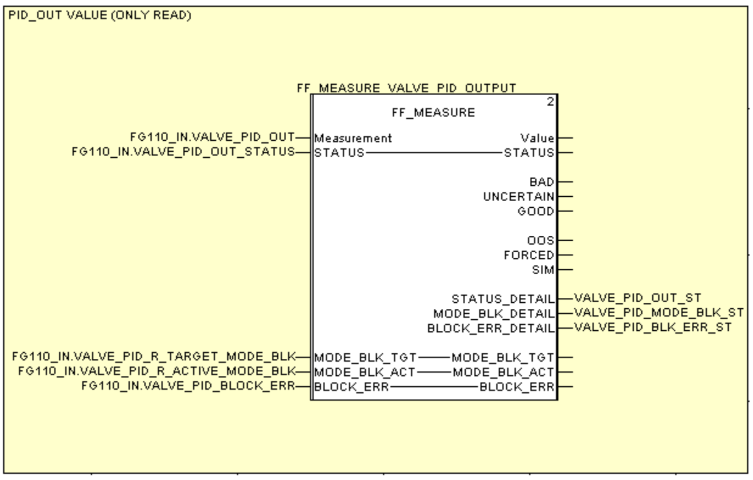


Figure 119: FF_MEASURE DFB used with the FOUNDATION Fieldbus PID

No specific Genie is used to show the PID output value because the same value is used for the input of the AO function block.

The *CVALVE* DFB of the DPL is used to control the valve on the PAC. This DFB is used in combination with the *FF_Measure* DFB to get the AO output value (valve position), the

MODE_BLK and the *BLOCK_ERR* parameters. The *Value* output of the *FF_Measure* DFB is linked to the *CHIN* input (feedback) of the *CVALVE* DFB and the remote function mode (see DPL user manual for more details) is activated for the *CVALVE* DFB in order to work with the *RSP* input.

The RSP is assigned to the PID output value obtained by the I/O Scanning.

Additionally, the *CONDSUM1* DFB of the DPL can be used to add several interlock conditions to the valve – in the application example there is no specific conditions added to the interlock.

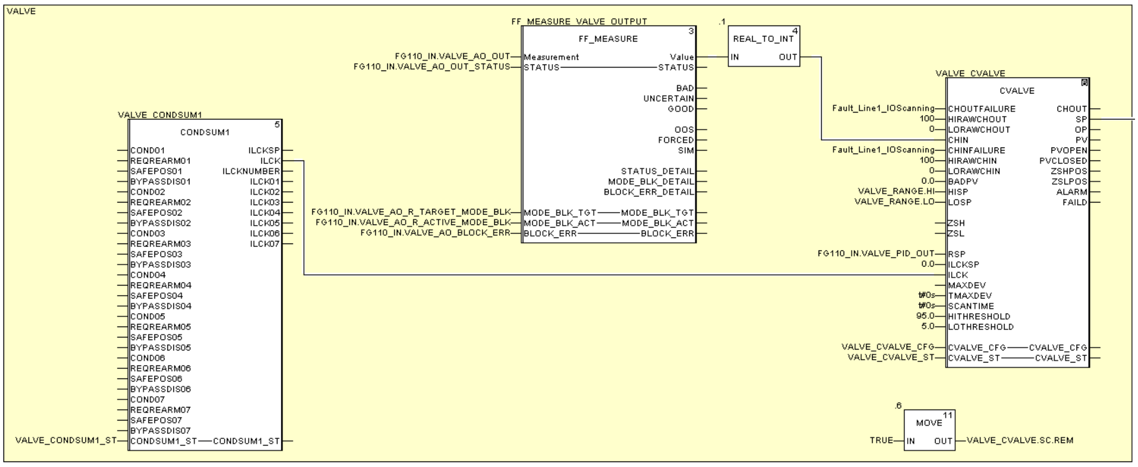


Figure 120: CVALVE, FF_MEASURE and CONDSUM1 DFBs for valve management

The program detailed previously can show the status of the output on the SCADA. To allow control using the generic Genie of the *CVALVE* DFB, we add some specific actions in case the *CVALVE* DFB is in manual mode, allowing the valve control to work independently of the FOUNDATION Fieldbus PID function block.

The additional programming part is detailed below:

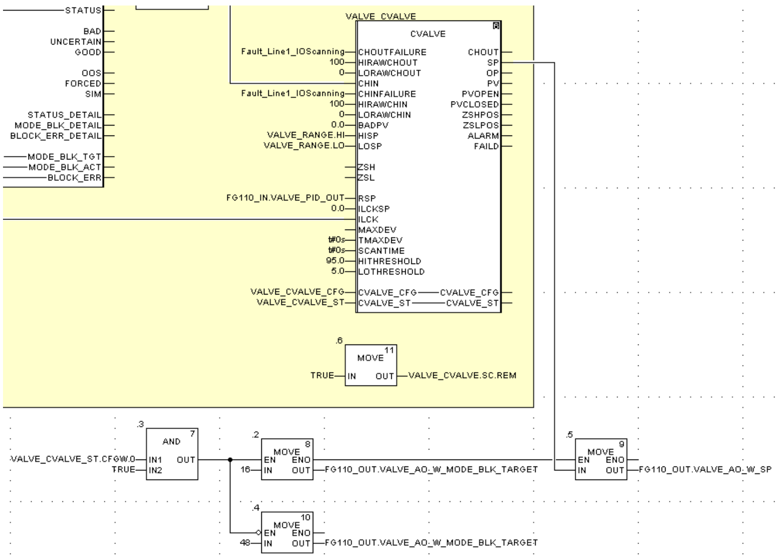


Figure 121: Valve manual management

If the control mode of the DFB is manual, the program assigns the value 16 (Manual) to the target mode of the FOUNDATION Fieldbus AO function block and it writes the setpoint of the *CVALVE* DFB to the FOUNDATION Fieldbus AO function block (the calculated value of the PID is not taken into account by the valve).

If the *CVALVE* DFB is running in automatic mode, the program writes the value 48 (corresponding to *Cascade+Automatic*) to the FOUNDATION Fieldbus AO function block to return to the normal state (taking into account the calculated value of the PID).

The *CVALVE* SuperGenie interface is shown below:

Figure 122: CVALVE SuperGenie used with the FOUNDATION Fieldbus PID

6.5. Function block diagnostics

The FOUNDATION Fieldbus function blocks provide two kinds of diagnostics information:

- The STATUS linked to an OUT value of the function block
- The BLOCK_ERR parameter of the function block

Specific means have been developed in the application example to use this information – DFBs for the PAC and Genies for the SCADA. The *FF_Measure* DFB provides specific structures to decode the information bit by bit.

The Genie specifically designed for the status information of a device is only meant to display the state of the different bits of the FOUNDATION Fieldbus measurement. The *FF_Measure* DFB is not mandatory on the PAC because this Genie can be connected directly to the DDT variable created by the NOC module.

As an example, the following figure shows the SuperGenie interface for AI measurement status of the H250 flow meter:

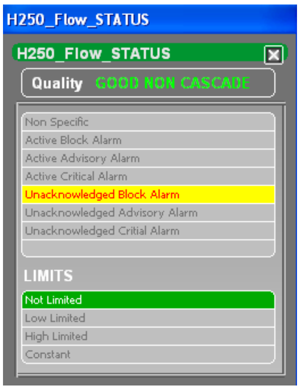


Figure 123: Measurement status SuperGenie

The Genie specifically designed for the *MODE_BLK* and the *BLOCK_ERR* is only meant to display the state of the different bits. Similarly, the usage of the *FF_Measure* DFB is not mandatory on the PAC because this genie can be connected directly to the DDT variable created by the NOC module.

The corresponding SuperGenie interface of the AI H250 flow meter is described below:

Figure 124: MODE_BLOCK and BLOCK_ERR status SuperGenie

6.6. Modbus gateway statistics

In order to monitor the statistics information available on the FG-110 linking device, we use the specific *FF_FG110_STATISTICS* DFB. The input parameter of this block is linked to the variables *FG110_IN.FG_110_Statistics* defined during the NOC configuration.

The figure below shows the parameters of the DFB as well as the Genie for the FG-110 in our project example:

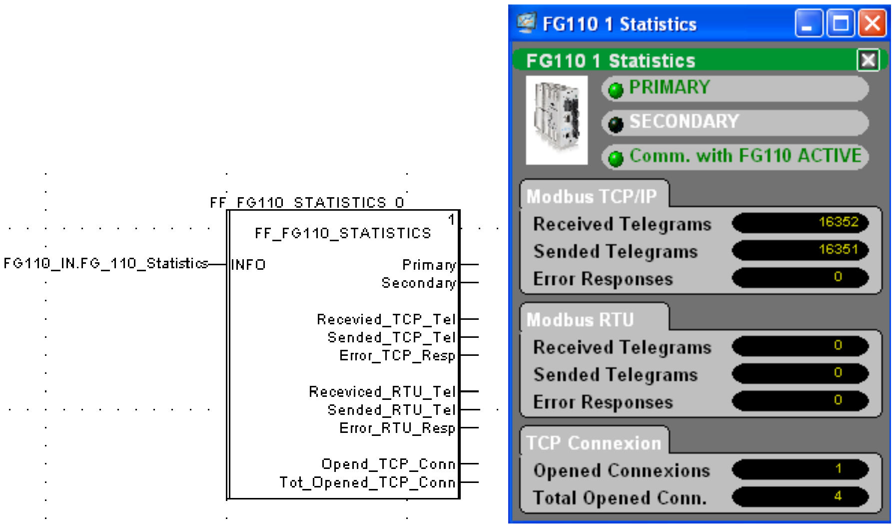


Figure 125: Modbus gateway DFB and SuperGenie

7. Operation and maintenance

This chapter provides operational examples of the SCADA system and asset management tools. The tools describe the use of the diagnostics and adjustment functions. They are designed for two use cases:

- An operator using the SCADA interface
- An instrumentation specialist using the asset management interface

⚠ WARNING
UNINTENDED EQUIPMENT OPERATION Configure your system depending on your own constraints (including environmental constraints). Test your system thoroughly before using it. **Failure to follow these instructions can cause death, serious injury or equipment damage.**

7.1. SCADA application example

The application example describes usage for a water tank, which uses the measurement equipment discussed in the previous chapters.

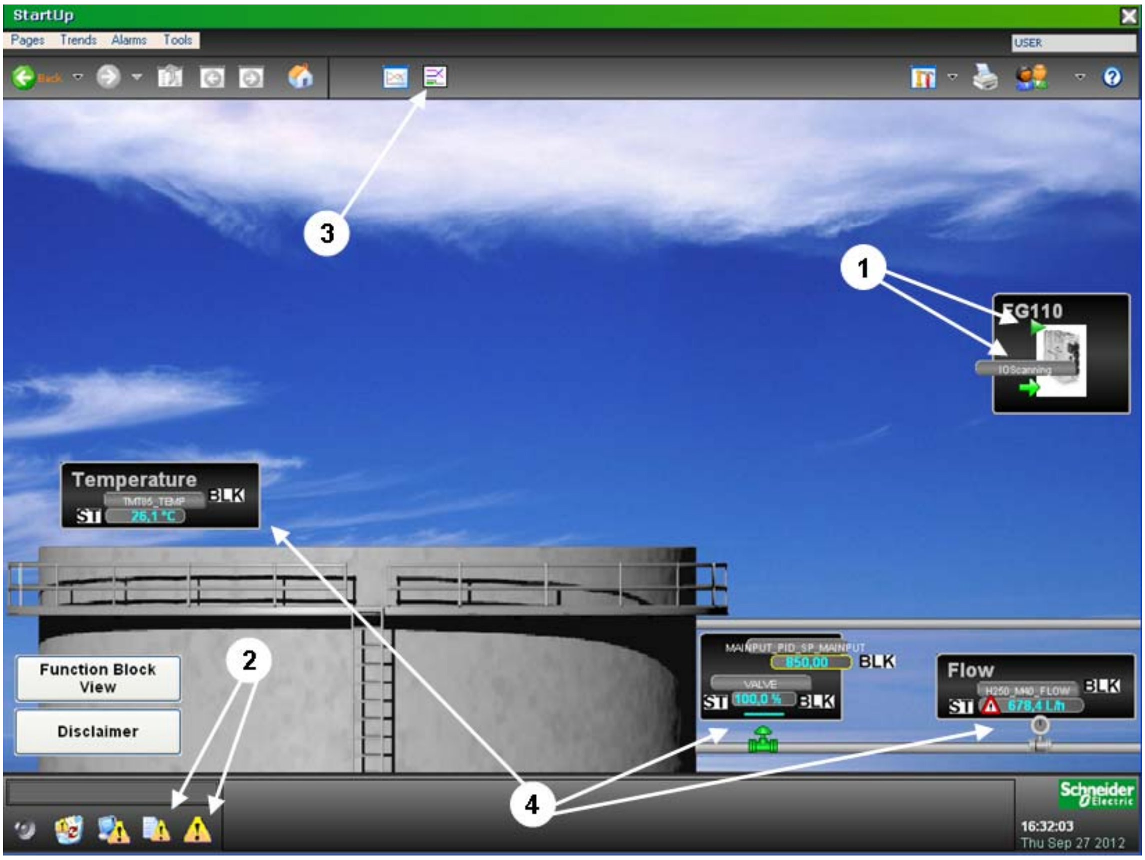


Figure 126: SCADA application example page

The items pointed in the capture above represent the following:

1. Communication status with the FG-110 linking device
2. Alarms
3. General measurement trends
4. Equipment status and measurement values

7.1.1. FG-110 statistics and communication status

You can supervise the communication status of the FG-110 linking device using two Genies:

- The specific Genie *FG110 Statistics* linked to the *FF_FG110_Statistics* DBF
- The DPL Genie for digital inputs *arrow_10* linked to the *DINPUT* DBF

With the *arrow_10* Genie, you can see the status of the I/O Scanning communication:

- The arrow is grey if the communication is not active and several blinking icons (bell and triangle) appear
- The arrow is green and not blinking when the communication is active

The *FG110 Statistics* Genie shows a green arrow when communication with the device is active and when the communication is not active the icon is blinking and red.

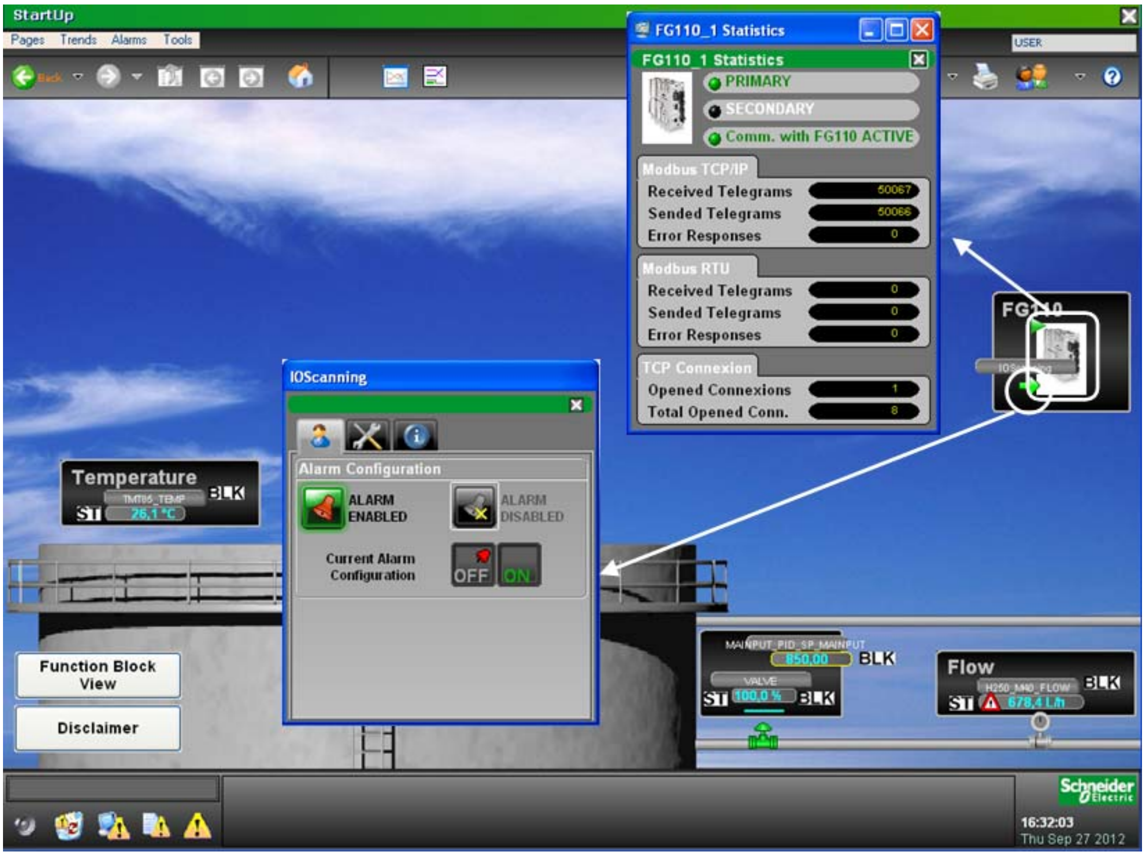


Figure 127: SCADA application example page with FG-110 Genies

7.1.2. Alarms

Specific links to pages with active alarms and the alarm summary are present in the template page used in the project. In addition, each Genie contains several symbols and color codes to note the specific alarms for the devices or functions to allow easy identification of any problem linked to the displayed page.

The table below describes the alarm symbols provided by the DPL Genies used in the application example:

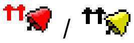
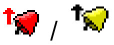
Icon	Description
	High-high level alarm
	High level alarm
	Low level alarm
	Low-low level alarm
	Channel failure* detected

Table 31: SCADA alarm icons

* The channel failure detection is managed by the *AINPUT* and *DINPUT* DBFs and is not linked to measurements status byte.

For each measurement instrument, a specific color provides an alarm status linked to the previous icons:

- Yellow: Alarm non-acknowledged (no longer active)
- Red: Channel failure detection or alarm active

For further information, please refer to the document *Device and Process Library – Citect Guide*.

7.1.3. Trends

In the template page used in the project, specific links to blank trending pages are included to customize trends. You can open trend pages from any page of the project application.

For each measured value, a trend display is available, as shown in the screenshot below:

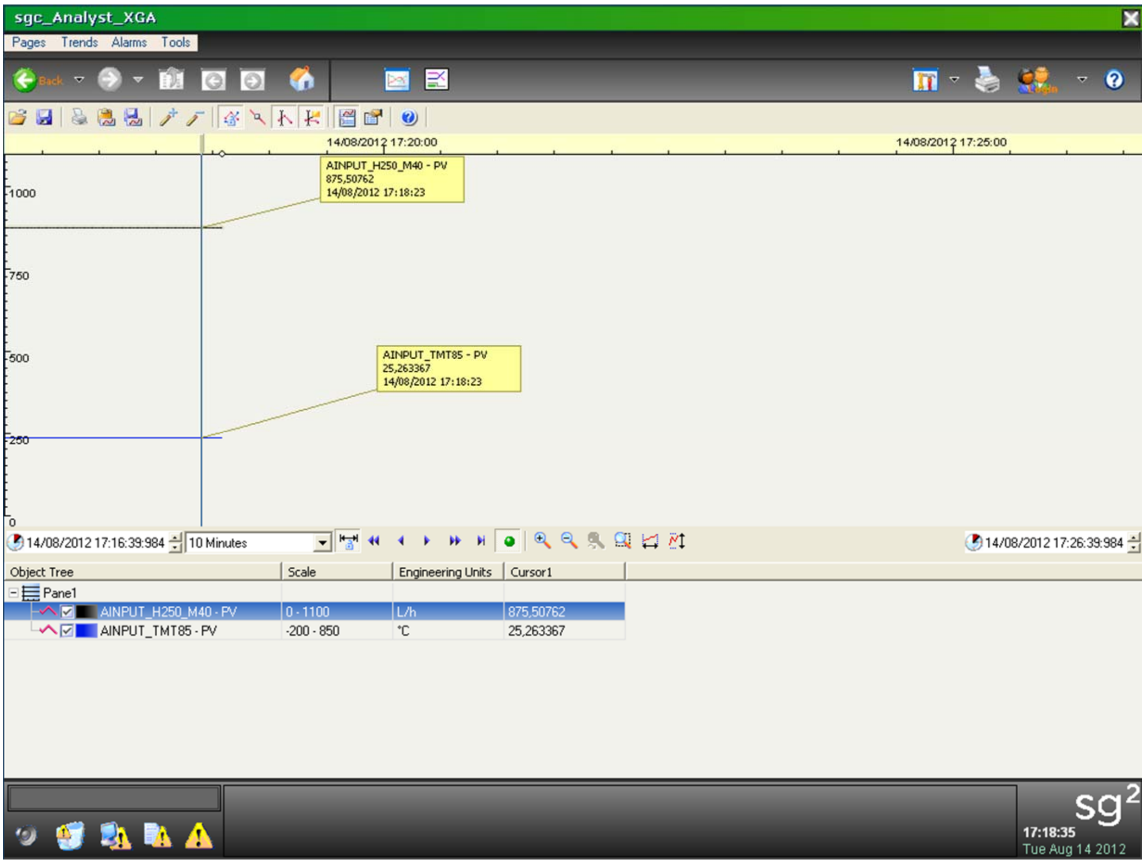


Figure 128: SCADA application example trending page

7.1.4. Instrument status and measurement

Each Genie used in this document provides a popup Super Genie with additional information. The following screenshot shows a SCADA page of the application developed for this document, including:

- The H250 flow meter with a non-acknowledged low-low alarm
- The H250 flow meter status byte details
- The H250 flow meter *MODE_BLK* and *BLOCK_ERR* details

Figure 129: SCADA application example with instrument genies

The information displayed in the first tab for the measurement instrumentation is:

- The current measured value (flow, temperature and so on, depending on the instrument)
- The scaled value of the measurement
- The four standard threshold values
- The setpoint (not used in our example)
- The deviation value for the setpoint, used to generate the alarm (not used in our example)

The other tabs contain information about the simulation mode.

The threshold adjustment is made directly in the faceplate of the Super Genie by completing the following procedure:

- Select the threshold value
- Type the new value
- Press the *Enter* key to validate

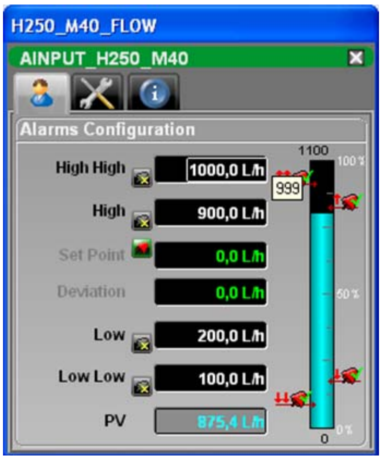


Figure 130: KROHNE flow meter SuperGenie detail

The measurement status SuperGenie displays the information of the FOUNDATION Fieldbus status. The sub-status information is automatically adapted by the SuperGenie, depending on the main status (*BAD*, *UNCERTAIN* and so on).

7.2. Asset management with Unity Pro

Asset management software allows the execution of several operations on devices, either online or offline. These operations can be performed using Unity Pro with the DTM container.

- *MODE_BLK* parameter set up
  - Resource block
  - Transducer block
  - Function block
- Device configuration and parameterization
  - Device tag, device type selection
  - Unit selection, date and time initialization
  - Device data allocation setup
  - Device calibration, scaling factors
  - Function block channel assignment
  - FOUNDATION Fieldbus alarming threshold values and reporting
- Device monitoring and control
  - Measurement tracking, actuator control
  - Simulated manipulations
- Device identification and diagnostics – standard and extended diagnostics data can be obtained for all instruments connected to a FOUNDATION Fieldbus H1 network

Note: The operation and features described above could vary depending on the DTM file provided by the manufacturer for a specific instrument.

The following table describes an example of a network discovery operation performed on the FOUNDATION Fieldbus H1 network. The Fieldbus discovery utility, available in the Unity Pro container, is used in this example.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Open the Unity Pro FDT container</td></tr>
<tr><td>2</td><td>
Right click on the FG-110 channel icon to open the contextual menu, and click *Connect*

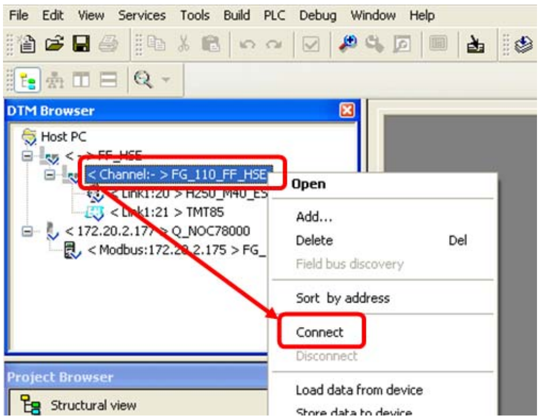


Figure 131: Unity Pro – connection to the FG-110 DTM
</td></tr>
<tr><td>3</td><td>
Once the Unity Pro FDT container is connected to the FG-110 linking device, the icon is displayed in bold

Right click again on the FG-110 channel icon to open the contextual menu, and click *Fieldbus discovery*

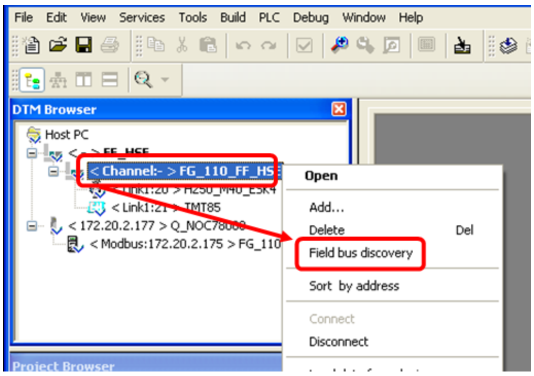
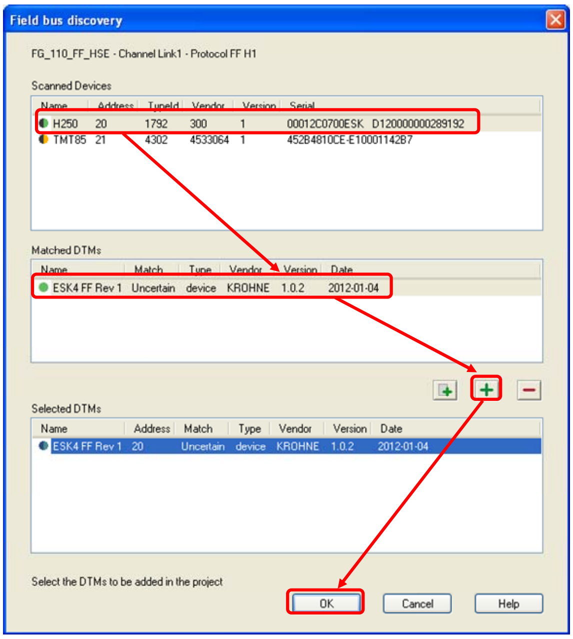


Figure 132: Unity Pro– fieldbus discovery
</td></tr>
</table>

Step	Action
4	A window appears and prompts you to choose the FG-110 H1 segment port which will perform the operation Choose the port and the following window appears with a list of the discovered devices:  **Figure 133: Unity Pro– discovered equipment list** Choose the device to be added and the corresponding DTM in the *Matched DTMs* list Click on the green + sign To finish the operation, click *OK* and the device will be displayed in the Unity DTM browser

Not all the DTMs provided by the manufacturers have the same behavior with the online mode. Some of them are permanently connected to the device (polling the device) when opened in online mode on the asset management station, while others do not perform any polling operation.

A good practice when identifying this behavior is to look at the left corner of the particular DTM once the device is in online mode. The figure below shows the DTM of the KROHNE H250 M40 ESK4:

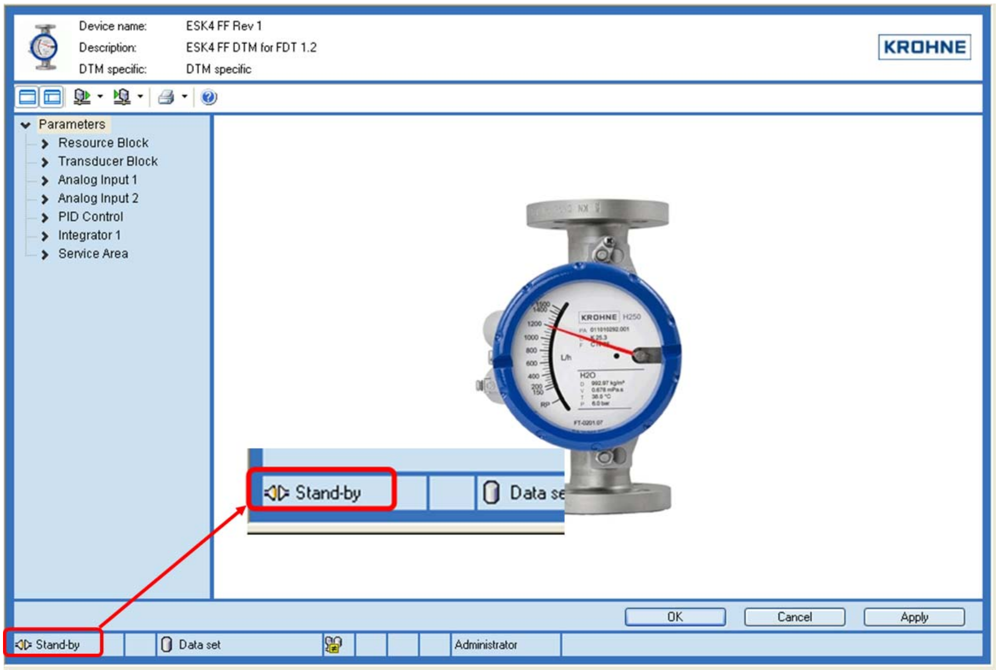


Figure 134: KROHNE H250 DTM

If the DTM does not execute any polling in online mode, you should use the menus intended to force the parameters read or write operations.

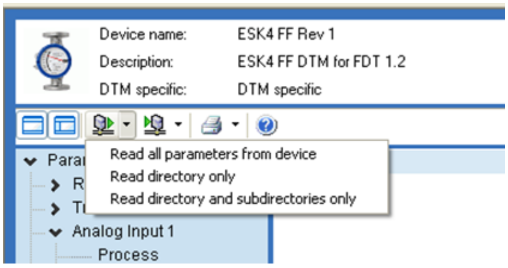


Figure 135: KROHNE H250 DTM read menu

7.3. FOUNDATION Fieldbus diagnostics

Three diagnostics levels are available when using the FG-110 linking device as a gateway for a control architecture based on Modbus communications:

- LAS diagnostics (FG-110 embedded web server)
- FOUNDATION Fieldbus status byte linked to the measurement (available in the web server, PAC and SCADA)
- FOUNDATION Fieldbus *BLOCK_ERR* for a particular function block (available in the PAC and SCADA)
- Instrumentation DTMs

7.3.1. FG-110 diagnostics

The FG-110 embedded web server provides several options for displaying diagnostics, including Modbus TCP/IP, Modbus RTU; FOUNDATION Fieldbus H1 and FOUNDATION Fieldbus HSE. Using the architecture proposed in this document, you can access the FG-110 web server (on the device network), using a standard web browser from the control network. The figure below shows an example of the information available in the FG-110 web server:

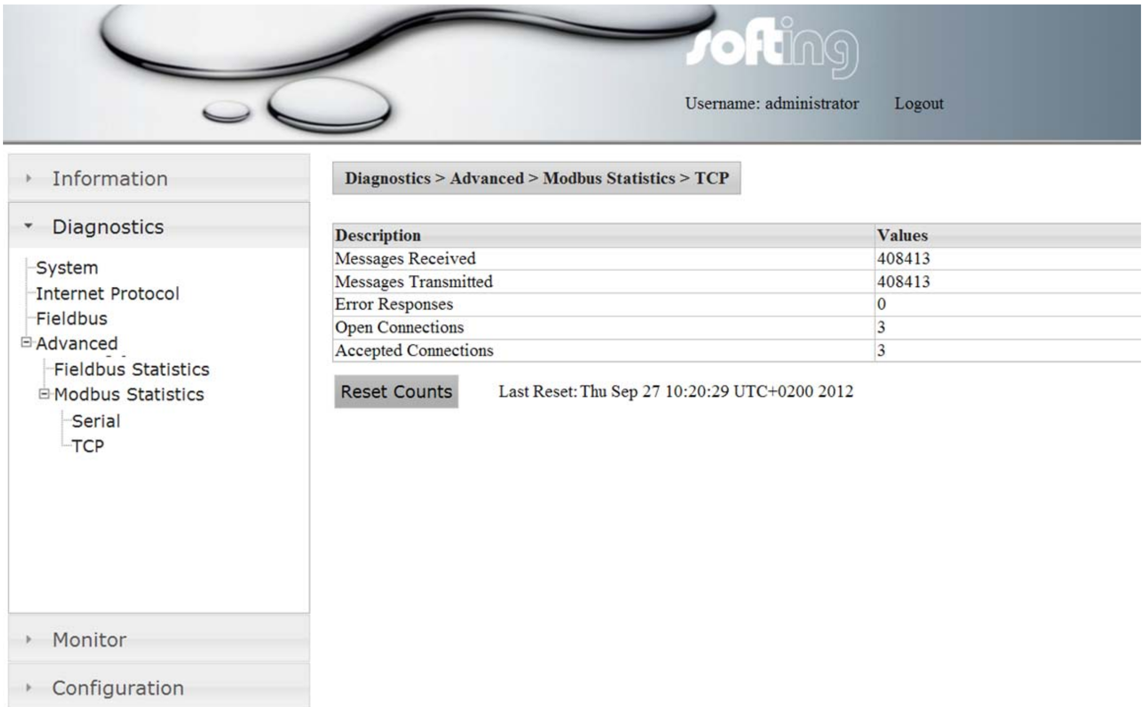


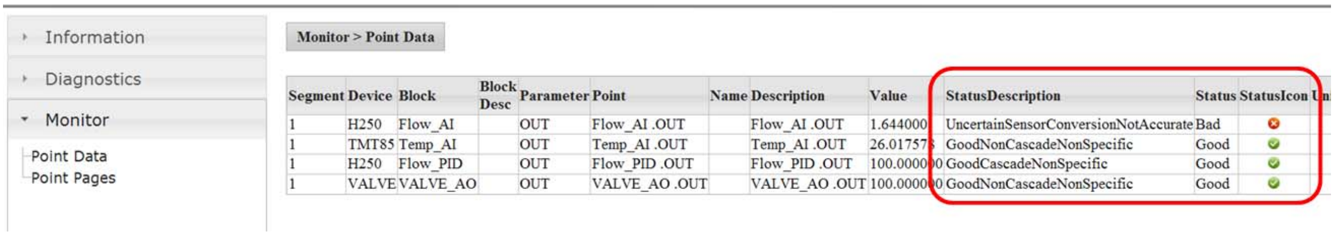
Figure 136: FG-110 web server diagnostics page

7.3.2. FOUNDATION Fieldbus measurement status byte

The FOUNDATION Fieldbus measurement status byte can be used as a first level detection for communication issues. The measurement status byte can be displayed using the following methods:

- Using the DFBs and Genies described in the subsection 7.1.4.
- Using the FG-110 embedded web server, declaring previously customized monitor pages

The figure below shows an example of the monitor page of the web server:



Segment	Device	Block	Block Desc	Parameter	Point	Name	Description	Value	StatusDescription	Status	StatusIcon	Un
1	H250	Flow_AI		OUT	Flow_AI .OUT		Flow_AI .OUT	1.644000	UncertainSensorConversionNotAccurate	Bad		
1	TMT85	Temp_AI		OUT	Temp_AI .OUT		Temp_AI .OUT	26.01757[illegible]	GoodNonCascadeNonSpecific	Good		
1	H250	Flow_PID		OUT	Flow_PID .OUT		Flow_PID .OUT	100.0000[illegible]0	GoodCascadeNonSpecific	Good		
1	VALVE	VALVE_AO		OUT	VALVE_AO .OUT		VALVE_AO .OUT	100.0000[illegible]0	GoodNonCascadeNonSpecific	Good		

Figure 137: FG-110 web server monitor page

For detailed information about the customized page creation, please refer to the FG-110 user manual.

If you need more detailed information about an issue, you can use the DTM as described in the next subsection.

7.3.3. Instrumentation DTMs

The FOUNDATION Fieldbus standard defines specific diagnostics information (block error), which must be detailed for each block included in the device (resource block, transducer blocks and function blocks). Furthermore, the manufacturer can include customized information about the specific diagnostics of the device.

The screenshots below show some of the diagnostics information available for the KROHNE H250 M40 flow meter.

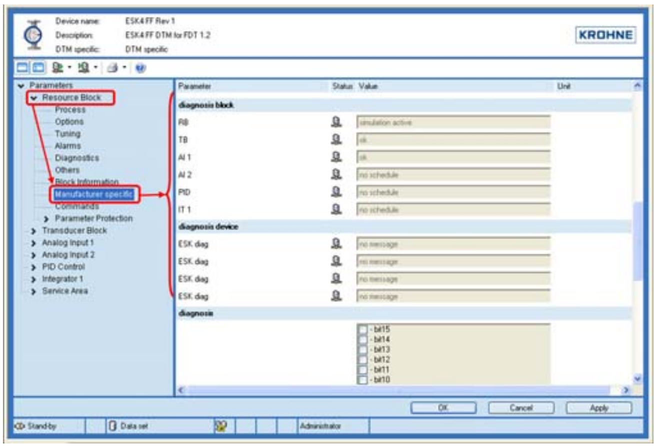


Figure 138: KROHNE H250 DTM resource block specific diagnostics

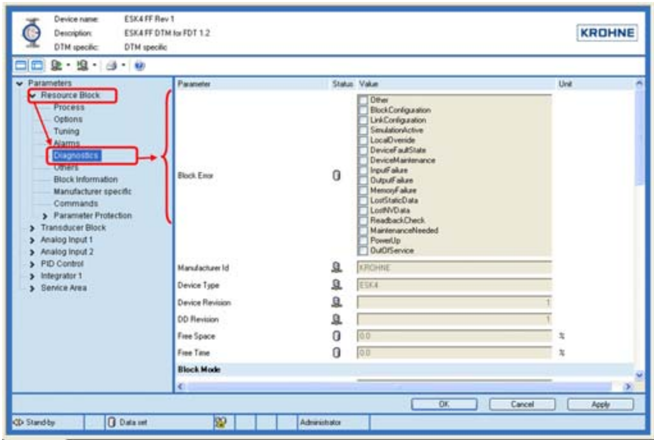


Figure 139: KROHNE H250 DTM resource block general diagnostics

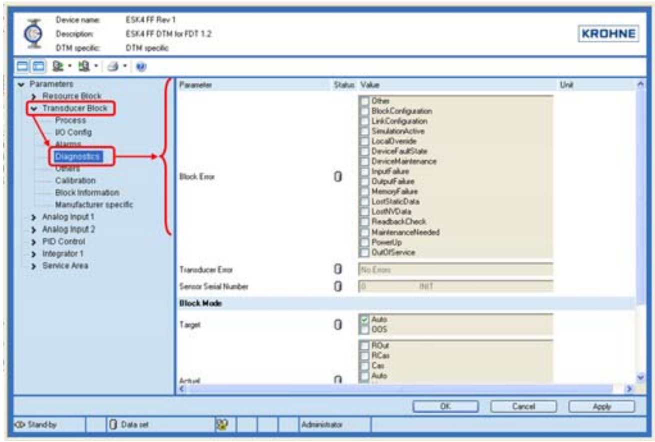


Figure 140: KROHNE H250 DTM transducer block diagnostics

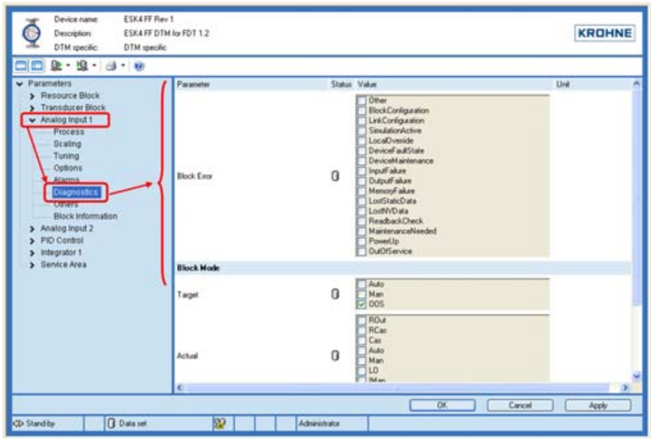


Figure 141: KROHNE H250 DTM analog input block diagnostics

Schneider Electric

8. Conclusion

Thanks to the FG-110 linking device, we can easily connect PlantStruxure to a FOUNDATION Fieldbus network. Monitoring and control of the field devices is simply achieved by mapping FOUNDATION Fieldbus function blocks and parameters to Modbus registers. When redundancy is required at the field bus level we can add a second linking device to act as a standby master.

All you need to setup the network is a web browser to map the Modbus information allowing diagnostics functionalities and a specific tool to configure the FOUNDATION Fieldbus network.

A major benefit of using a gateway device is minimizing the cost of the H1 cabling, allowing distributed architectures thanks to the Ethernet connectivity from the PAC to the linking device. Remote diagnostics and maintenance are greatly facilitated.

The FG-110 linking device uses DTM technology in line with the PlantStruxure technical strategy. Combined with the Ethernet connectivity, we can offer a high integration level between the PlantStruxure architectures and FOUNDATION Fieldbus devices. The asset management can be executed directly with Unity Pro (which includes a FDT container). The process data values are automatically created in the Unity Pro project with the adequate format (real, integer, and so on) thanks to the DTM technology.

9. Appendix

9.1. Glossary

The following table describes the acronyms and defines the specific terms used in this document.

Term	Description
AI	Analog Input
AO	Analog Output
CD	Compel Data
CFF	Common File Format
DD	Device Description
DDL	Device Description Language
DDT	A Derived Data Type is a data structure in Unity Pro
DI	Digital Input
DLL	Data Link Layer
DO	Digital Output
DPL	Device and Process Libraries propose function blocks for Unity Pro and the corresponding SCADA objects for Vijeo Citect
DTM	The Device Tool Manager provides a unified structure for accessing device parameters, configuring and operating the devices and diagnosing problems. DTMs can range from a simple Graphical User Interface for setting device parameters to a highly sophisticated application capable of performing complex real-time calculations for diagnostics and maintenance purposes
FAS	Fieldbus Access Sub-layer
FDT	The Field Device Technology standardizes the communication and configuration interface between all field devices and host systems. FDT provides a common environment for accessing the devices' most sophisticated features. Any device can be configured, operated and maintained through the standardized user interface – regardless of supplier, type or communication protocol
FF	FOUNDATION Fieldbus
FFB	Flexible Function Block
FFPS	FOUNDATION Fieldbus Power Supply

Term	Description
FF-SIS	FOUNDATION Fieldbus for Safety Instrumented Systems
FISCO	Fieldbus Intrinsically Safe Concept
FMS	Fieldbus Message Specification
HSE	FOUNDATION Fieldbus High Speed Ethernet
IEC	International Electrotechnical Commission
IPF	Instrumented Protective Function
IS	Intrinsic Safety
LAS	Link Active Schedule
LM	Link Master
MBP	Manchester-encoded, Bus powered
NAMUR	“Normenarbeitsgemeinschaft für Meß- und Regeltechnik in der chemischen Industrie” – Instrumentation standard for the chemical industry
PAC	A Programmable Automation Controller is a high-end PLC with advanced services and capabilities.
PN	Probe Node
PR	Probe Response
PT	Pass Token
RB	Resource Block
SIF	Safety Instrumented Function
SIL	Safety Integrity Level
SIS	Safety Instrumented System
Spur	Derivation of the main fieldbus cable
TB	Transducer Block
TD	Time Distribution
Trunk	Main fieldbus cable
VCR	Virtual Communication Resource
VFD	Virtual Field Devices

Table 32: glossary

9.2. Bill of material and software

The following table summarizes the selected hardware:

Description	Reference	Firmware or software version	Function
QUANTUM	140XBP00600 140CPS11420 140CPU65150	- - OS V3.10 / Cop V3.80	Quantum rack, 6 slot Quantum PAC power supply Quantum PAC
Quantum NOC	140NOC78000 PSx EIP Master Generic Modbus	V1.56 – IE03- 1.2.18.0 V1.1.10.0	Ethernet NOC DIO module NOC DTM Configuration file Generic Modbus Device DTM file
FG-110-FF	ICM-NK-0211 FF HSE/LD	V1.70.0.08 V1.21(64)	FF Linking Device – Modbus Gateway Softing DTM Linking Device
Junction Box	8411/24-310-41	-	FF junction box from STAHL
FF Power Supply	9412/00-310-11	-	Specific FF Power Supply from STAHL
KROHNE H250 flow meter	H250/M40/ES4 ESK4 010101	- V1.0.2 V1.0 First Release	KROHNE variable area flow meter ESK4 device DTM EDD file H250 ESK4-FF
E+H temperature sensor	TMT85 DDCFFEH_000Lib 010105	- V1.6.40.106 V1.7	E+H temperature sensor TMT85 device DTM EDD file iTEMP TMT85 FF
Samson valve positioner	3730-5 SAMSON 3730-5 020101	K2.00/R1.52 V K2.00 V1.7	Electropneumatic valve positioner comm. 3730-5 device DTM EDD file 3730-5
FF Design	DesignMATE	V1.1.118	Planning and verification of the FOUNDATION Fieldbus H1 segments.
Unity Pro XL	UNYSPUEFFCD70	7.0	Schneider configuration tool for PAC
OFS	TLXCDLUOFS35	V3.35	OPC data server software for single station

Description	Reference	Firmware or software version	Function
Vijeo Citect	VJC109922 VJCNS101114	7.20 SP2	Hardware delivery of DVD / USB key Server license for 5000 points
Softing FF Conf. tool	FF-CONF	V1.1.97.0	Softing FF configuration tool for FG-110 module
Web Browser	Windows Internet Explorer	8.0.6001.18702	Microsoft Internet Explorer
DPL		V1.0	Device Process Library for PlantStruxure

Table 33: bill of material and software

9.3. Reference documents

The following table is a list of documents you might want to refer to when more details are needed.

Document title	Reference
FF - System Engineering Guidelines	AG-181 - Revision 3.1
FF - Tech Overview	FD-043 Rev 3.0
FF - Wiring Installation Guide	AG140 – Revision 1.0
FF – Intrinsically Safe Systems	AG163 – Revision 2.0
FF Specification Function Block Application Process (Part 1)	FF-890
FF Specification Function Block Application Process (Part 2)	FF-891
FF Specification Function Block Application Process (Part 3)	FF-892
FF Specification Function Block Application Process (Part 4)	FF-893
SAMSON FF Technical Information	L454EN
E+H - Operating Instructions iTEMP TMT85	BA251R/09/en/10.07
E+H - Safety Instructions Temperature Transmitters iTEMP	XA01006T/09/a3/02.12
E+H - Technical Information iTEMP TMT85	TI00134R/09/en
KROHNE – H250 M40 Handbook	4000640702
KROHNE – M250 M40 Description of FF interface	4001208201

Document title	Reference
STAHL – Operating Instructions Field Device Coupler (8 spurs)	941160310190
STAHL – Operating Instructions FF Power Supply System	941260310030
SOFTING Linking Device Manual V1.1 (FG-110)	-
Quantum EIO – Installation and Configuration Guide (NOC)	S1A48986.00
DPL (V1.0) - Process Function Blocks - User Manual	-
DPL (V1.0) - Vijeo Citect Components - User Manual	-

Table 34: Reference documents

The following websites provide additional information about FOUNDATION Fieldbus:

http://www.fieldbus.org/

http://www.eddl.org/

Schneider Electric Industries SAS
Head Office
35, rue Joseph Monier
92506 Rueil-Malmaison Cedex
FRANCE

www.schneider-electric.com

Due to evolution of standards and equipment, characteristics indicated in texts and images in this document are binding only after confirmation by our departments.

Print:

Version 1.00 – 11 2012